

# Important Notice

This copy may be used only for the purposes of research and private study, and any use of the copy for a purpose other than research or private study may require the authorization of the copyright owner of the work in question. Responsibility regarding questions of copyright that may arise in the use of this copy is assumed by the recipient.

UNIVERSITY OF CALGARY

Classification of Microseismic Events from Bitumen Production at Cold Lake, Alberta

by

Jeffrey Francis Tan

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF GEOSCIENCE

CALGARY, ALBERTA

DECEMBER, 2007

© Jeffrey Francis Tan 2007

UNIVERSITY OF CALGARY  
FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled “Classification of Microseismic Events from Bitumen Production at Cold Lake, Alberta” submitted by Jeffrey Francis Tan in partial fulfilment of the requirements of the degree of Master of Science.

---

*Supervisor, Dr. R.R. Stewart  
Department of Geoscience*

---

*Dr. G.F. Margrave  
Department of Geoscience*

---

*Dr. A.O. Fajoluwo  
Department of Electrical and Computer Engineering*

---

*Date*

## **ABSTRACT**

This work develops and tests microseism event-classification techniques. Research was performed in collaboration with CREWES using microseismic data from Cold Lake, Alberta that was provided by Imperial Oil Ltd. The objective was to develop passive-seismic signal classification algorithms capable of precisely and automatically distinguishing between microseismic events warranting further investigation from noise events that are generally not of interest. Novel methods involving frequency-filtering, event-length detection, and statistical analysis were developed.

After extensive testing, it was found that developed statistical analysis algorithms performed best. Principal components analysis was applied to statistical analysis algorithm outputs to optimize classification.

Two MATLAB<sup>®</sup> implementation schemes were created. The second, most recent, application yielded classification accuracies between 90% and 99.5% when tested on a wide range of datasets. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this work could have significant future impact.

## ACKNOWLEDGEMENTS

Thanks to my supervisor, Dr. Robert Stewart, for giving me the opportunity to meaningfully contribute to CREWES by allowing me to work on the Cold Lake microseismic monitoring project. Rob allowed an open project scope, which gave me the chance to be creative in providing improved solutions to a real world problem. He certainly put me in a position where I had a chance to succeed.

Henry Bland, a former CREWES staff member, was very helpful in providing ideas, organizing meetings with Imperial Oil employees, and providing microseismic data. He is now employed with Pinnacle Technologies, where I wish him success.

Colum Keith and Richard Smith from Imperial Oil were the ones who made me aware about the Cold Lake event-classification issues. Their feedback was always insightful and constructive. Sophia Follick, who was an internship student at Imperial, performed a lot of testing on my program and was punctual in providing testing results.

Dr. Gary Margrave gave me insight on how to write quality technical papers and taught me the value of including technical and mathematical content. I learned a lot in his GOPH 557 processing course, which was excellent and very well-taught.

Thanks to all CREWES staff, students, and sponsors. Louise Forgues is an excellent administrator. Kevin Hall saved me hours when I had computer issues. There are too many people to name here. It was a pleasure working with all of them when I had a chance.

Most importantly, thanks to my family: Mom, Dad, Dave, Jackie, and Lucas for all their support in my life.

I have enjoyed my time as a CREWES grad student. Having the opportunity to present this work at the CREWES sponsors meeting in addition to the CSEG and SEG conferences were new and worthwhile experiences.

Being a part of CREWES has opened up innumerable opportunities. I now have key technical and industry perspective, which I plan to continually improve upon and apply to the full extent in the future.

## **DEDICATION**

To Mom, Dad, David, Jacklyn, and Lucas

## TABLE OF CONTENTS

APPROVAL PAGE.....	ii
ABSTRACT.....	iii
ACKNOWLEDGEMENTS.....	iv
DEDICATION.....	v
TABLE OF CONTENTS.....	vi
LIST OF TABLES .....	ix
LIST OF FIGURES.....	xi
LIST OF SYMBOLS AND ABBREVIATIONS.....	xxiv
CHAPTER ONE: INTRODUCTION .....	1
1.1 Passive-Seismic Monitoring.....	1
1.2 Cold Lake Oil Sands.....	2
1.2.1 Background.....	2
1.2.2 Imperial Oil Operations at Cold Lake.....	3
1.2.3 Passive-Seismic Monitoring Requirement.....	5
1.2.3.1 Microseismic Events.....	9
1.3 Synthetic Data.....	21
1.3.1 “Good” Events.....	21
1.3.2 Noise Event.....	25
1.4 Thesis Objective.....	26
1.5 Original Thesis Contributions.....	26
1.6 Thesis Outline.....	28
CHAPTER TWO: FREQUENCY FILTERING.....	30
2.1 Introduction.....	30
2.2 Background.....	30
2.2.1 Terminology.....	30
2.2.2 Common Filter Frequency Response Approximations.....	31
2.3 Low-Pass Inverse Chebyshev Filter.....	34

2.4 High-Pass Butterworth Filter.....	43
2.5 Band-Pass Chebyshev Filter.....	52
2.6 Conclusion.....	60
CHAPTER THREE: EVENT-LENGTH DETECTION.....	61
3.1 Introduction.....	61
3.2 STA / LTA Algorithm.....	62
3.3 Continuous-Time Frequency Analysis.....	69
3.4 Conclusion.....	86
CHAPTER FOUR: STATISTICAL ANALYSIS.....	87
4.1 Introduction.....	87
4.2 Statistical Characteristics of Microseismic Signals.....	87
4.2.1 Signal Variance.....	88
4.2.1.1 Chebyshev's Inequality.....	90
4.2.2 Signal Distribution.....	93
4.2.3 Signal Oscillation and Sequential Time-Series Behavior.....	97
4.3 Statistical Classification Algorithms.....	99
4.3.1 Threshold Algorithm.....	99
4.3.2 Histogram Algorithm.....	101
4.3.3 Specialized Zero-Crossing Count Algorithm.....	109
4.4 Conclusion.....	112
CHAPTER FIVE: MULTIVARIATE DATA REDUCTION.....	113
5.1 Introduction.....	113
5.2 Principal Components Analysis.....	114
5.3 Test Results.....	115
5.4 Applying PCA to Microseismic File Classification.....	121
5.4.1 Applying PCA to All Eight Algorithm Outputs.....	121
5.4.2 Restricting PCA to Statistical Algorithm Outputs.....	129
5.5 Conclusion.....	134

CHAPTER SIX: ALGORITHM IMPLEMENTATION.....	135
6.1 Introduction.....	135
6.2 Program Implementations.....	136
6.2.1 Graphical User Interface.....	136
6.2.2 Principal Components Analysis Applied.....	139
6.3 Results.....	142
6.4 Conclusion.....	143
CHAPTER SEVEN: CONCLUSIONS AND FUTURE WORK.....	144
7.1 Summary.....	144
7.2 Future Work.....	146
Appendix A.....	147
Appendix B.....	152
Appendix C.....	157
REFERENCES.....	169

## LIST OF TABLES

Table 2.1: Summary of peak amplitudes of synthetic traces after applying low-pass Inverse-Chebyshev filter.....	35
Table 2.2: Summary of peak amplitudes of “good” and noise events after applying low-pass Inverse-Chebyshev filter. Note that the magnitudes of the filtered “good” events are considerably higher than those of the filtered noise events.....	36
Table 2.3: Summary of peak amplitudes of synthetic traces after applying high-pass Butterworth filter.....	45
Table 2.4: Summary of peak amplitudes of “good” and noise events after applying high-pass Butterworth filter. Note that the magnitudes of the filtered “good” events are considerably lower than those of the filtered noise events.....	45
Table 2.5: Summary of peak amplitudes of synthetic traces after applying band-pass Chebyshev filter.....	53
Table 2.6: Summary of peak amplitudes of “good” and noise events after applying band-pass Chebyshev filter. Note that the magnitudes of the filtered “good” events are generally considerably lower than those of the filtered noise events.....	54
Table 3.1: Summary of detected event-lengths using STA / LTA technique. Note that event-lengths of example “good” events are considerably shorter than those of example noise events.....	68
Table 3.2: Summary of detected event-lengths. Note that event-lengths of example “good” events are considerably shorter than those of example noise events.....	78
Table 4.1: Summary of the percentage of outlying data points using a single-sided window width of $a = 0.03$ when Threshold algorithm is applied to the synthetic traces....	100
Table 4.2: Summary of the percentage of outlying data points using a single-sided window width of $a = 0.03$ when Threshold algorithm is applied to example “good” and noise events. Note that, compared to noise traces, the example “good” traces generally contain fewer outlying data points.....	100
Table 4.3: Summary of the percentage of data points in the 50 <sup>th</sup> (middle) bin range when the Histogram algorithm is applied to the synthetic traces.....	102

Table 4.4: Summary of the percentage of data points in the 50 <sup>th</sup> (middle) bin range when the Histogram algorithm is applied to example “good” and noise events. Note that, compared to “good” events, noise events generally have fewer data points that lie in the 50 <sup>th</sup> bin range.....	102
Table 4.5: Summary of the percentage of valid polarity reversals (zero-crossings) when the Specialized Zero-Crossing Count algorithm is applied to the synthetic traces.....	110
Table 4.6: Summary of the percentage of valid polarity reversals (zero-crossings) when the Specialized Zero-Crossing Count algorithm is applied to example “good” and noise events. Note that “good” events generally have significantly less valid polarity reversals than noise events.....	111
Table 6.1: Performance of example traces. Trace values that "pass" are in boldface... ..	138

## LIST OF FIGURES

Figure 1.1: Alberta’s three major oil sands deposits. The Cold Lake deposit is located in the east-central part of the province (Imperial Oil Ltd., 2006a).....	2
Figure 1.2: Imperial’s Cold Lake operations (Imperial Oil Ltd., 2006b).....	3
Figure 1.3: Simplified stratigraphic chart of Cold Lake area (Isaac, 1996).....	4
Figure 1.4: CSS enhanced oil recovery process for a single production well. The well is put through cycles of steam injection, soaking, and production, which is repeated so long as it is economically viable (Imperial Oil Ltd.,2006c).....	5
Figure 1.5: Example schematic of a five-geophone array lowered into a monitoring well to detect microseismic events occurring in production wells. When a microseism occurs, elastic P- and S-waves are propagated from its focus. The focus in this example occurs in the Grand Rapids formation, above the producing Clearwater formation.....	7
Figure 1.6: Example outline of a microseismic file assuming geophone configuration depicted in Figure 1.5. As each 3-C geophone outputs three traces, one for each component ( $x$ , $y$ , and $z$ ), this example file contains fifteen traces of microseismic activity. Each trace in this example has a temporal duration of 1.365 seconds. The three traces at the top of a microseismic event file come from the shallowest geophone; the next three traces below are from the second-shallowest geophone, and so on. Each trace is represented by an arrow in this figure.....	8
Figure 1.7: Example trace of a “good” event with minimal superimposed noise. The impulsive P- and S-wave arrival signatures indicate the presence of a “good” event. This “good” event is labeled “Event #1” for future reference.....	15
Figure 1.8: A “good” event with high-frequency noise superimposed. The P-wave arrival is still easily distinguishable and is followed by a significantly larger S-wave arrival. This “good” event is labeled “Event #2” for future reference.....	15
Figure 1.9: A “good” event where a very impulsive, high-amplitude P-wave arrival is followed by a lower frequency, low-amplitude S-wave arrival. This amplitude decrease could be due to fluid content between the event hypocenter and the sensors and differences between subsurface compressional and shear quality factors. This “good” event is labeled “Event #3” for future reference.....	16
Figure 1.10: A “good” event with an impulsive P-wave arrival followed by an impulsive S-wave arrival of higher amplitude. Low to moderate frequency noise is seen in this event, but P- and S-wave arrivals are still distinct and easily distinguishable. This “good” event is labeled Event #4 for future reference.....	16

Figure 1.11: A “good” event with an impulsive P-wave arrival followed by an S-wave arrival with lower amplitude and lower frequency. Noise is negligible in this trace. This “good” event is labeled Event #5 for future reference.....	16
Figure 1.12: A “good” event with a P-wave arrival followed by a much larger S-wave arrival. Superimposed low-frequency noise is seen. This “good” event is labeled Event #6 for future reference.....	17
Figure 1.13: A “good” event with comparable P- and S-wave arrival amplitudes. Noise occurs between the P- and S-wave arrivals. This “good” event is labeled Event #7 for future reference.....	17
Figure 1.14: A “good” event with an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and low-frequency noise. This “good” event is labeled Event #8 for future reference.....	17
Figure 1.15: A “good” event with an impulsive P-wave arrival followed by noise with wideband frequency components, an S-wave arrival of smaller amplitude, and low-frequency noise. This “good” event is labeled Event #9 for future reference.....	18
Figure 1.16: A “good” event with an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and negligible noise thereafter. This “good” event is labeled Event #10 for future reference.....	18
Figure 1.17: Example trace of a noise event. This event could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #11 for future reference.....	19
Figure 1.18: A noise trace containing an event of long duration. This event could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #12 for future reference.....	19
Figure 1.19: A noise trace containing two events of long duration. These events could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #13 for future reference.....	20
Figure 1.20: A noise trace containing a relatively impulsive, high frequency noise event, possibly caused by high-frequency (short duration) pump rod oscillation in production wells. This noise event is labeled Event #14 for future reference.....	20
Figure 1.21: Random, lower frequency noise. This is likely random jitter detected by the sensors, as noise due to oscillating pump rods and passing vehicles tend to cause microseisms of higher frequencies. This noise event is labeled Event #15 for future reference.....	20

Figure 1.22: Synthetic “good” trace with SNR = 30 dB.....	23
Figure 1.23: Synthetic “good” trace with SNR = 25 dB.....	24
Figure 1.24: Synthetic “good” trace with SNR = 20 dB.....	24
Figure 1.25: Synthetic “good” trace with SNR = 15 dB.....	24
Figure 1.26: Synthetic “good” trace with SNR = 10 dB.....	24
Figure 1.27: Synthetic “good” trace with SNR = 0 dB.....	25
Figure 1.28: Synthetic noise trace.....	25
Figure 2.1: Sketches of attenuation characteristics of 4 main filter responses for a low-pass filter case: Butterworth (top-left), Chebyshev (top-right), Inverse Chebyshev (bottom-left), and Cauer (bottom-right). The horizontal axis $\omega$ pertains to angular frequency measured in radians per second. The vertical axis, $\alpha(\omega)$ , represents the filter’s loss function measured in decibels (dB) (adapted from Maundy, 2005).....	32
Figure 2.2: Amplitude response of low-pass Inverse-Chebyshev filter.....	34
Figure 2.3: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.21 with SNR = 30 dB.....	36
Figure 2.4: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 25 dB.....	36
Figure 2.5: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 20 dB.....	37
Figure 2.6: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 15 dB.....	37
Figure 2.7: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 10 dB.....	37
Figure 2.8: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 0 dB.....	38
Figure 2.9: Low-pass Inverse-Chebyshev filter applied to synthetic noise trace shown in Figure 1.27.....	38
Figure 2.10: Low-pass Inverse-Chebyshev filter applied to Event #1, a “good” event.....	38
Figure 2.11: Low-pass Inverse-Chebyshev filter applied to Event #2, a “good” event.....	39

Figure 2.12: Low-pass Inverse-Chebyshev filter applied to Event #3, a “good” event.....	39
Figure 2.13: Low-pass Inverse-Chebyshev filter applied to Event #4, a “good” event.....	39
Figure 2.14: Low-pass Inverse-Chebyshev filter applied to Event #5, a “good” event.....	40
Figure 2.15: Low-pass Inverse-Chebyshev filter applied to Event #6, a “good” event.....	40
Figure 2.16: Low-pass Inverse-Chebyshev filter applied to Event #7, a “good” event.....	40
Figure 2.17: Low-pass Inverse-Chebyshev filter applied to Event #8, a “good” event.....	41
Figure 2.18: Low-pass Inverse-Chebyshev filter applied to Event #9, a “good” event.....	41
Figure 2.19: Low-pass Inverse-Chebyshev filter applied to Event #10, a “good” event.....	41
Figure 2.20: Low-pass Inverse-Chebyshev filter applied to Event #11, a noise event.....	42
Figure 2.21: Low-pass Inverse-Chebyshev filter applied to Event #12, a noise event.....	42
Figure 2.22: Low-pass Inverse-Chebyshev filter applied to Event #13, a noise event.....	42
Figure 2.23: Low-pass Inverse-Chebyshev filter applied to Event #14, a noise event.....	43
Figure 2.24: Low-pass Inverse-Chebyshev filter applied to Event #15, a noise event.....	43
Figure 2.25: Amplitude response of high-pass Butterworth filter.....	44
Figure 2.26: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.21 with SNR = 30 dB.....	46
Figure 2.27: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 25 dB.....	46
Figure 2.28: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 20 dB.....	46
Figure 2.29: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 15 dB.....	47
Figure 2.30: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 10 dB.....	47
Figure 2.31: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 0 dB.....	47

Figure 2.32: High-pass Butterworth filter applied to synthetic noise trace shown in Figure 1.27.....	48
Figure 2.33: High-pass Butterworth filter applied to Event #1, a “good” event.....	48
Figure 2.34: High-pass Butterworth filter applied to Event #2, a “good” event.....	48
Figure 2.35: High-pass Butterworth filter applied to Event #3, a “good” event.....	49
Figure 2.36: High-pass Butterworth filter applied to Event #4, a “good” event.....	49
Figure 2.37: High-pass Butterworth filter applied to Event #5, a “good” event.....	49
Figure 2.38: High-pass Butterworth filter applied to Event #6, a “good” event.....	49
Figure 2.39: High-pass Butterworth filter applied to Event #7, a “good” event.....	50
Figure 2.40: High-pass Butterworth filter applied to Event #8, a “good” event.....	50
Figure 2.41: High-pass Butterworth filter applied to Event #9, a “good” event.....	50
Figure 2.42: High-pass Butterworth filter applied to Event #10, a “good” event.....	50
Figure 2.43: High-pass Butterworth filter applied to Event #11, a noise event.....	51
Figure 2.44: High-pass Butterworth filter applied to Event #12, a noise event.....	51
Figure 2.45: High-pass Butterworth filter applied to Event #13, a noise event.....	51
Figure 2.46: High-pass Butterworth filter applied to Event #14, a noise event.....	51
Figure 2.47: High-pass Butterworth filter applied to Event #15, a noise event.....	52
Figure 2.48: Amplitude response of band-pass Chebyshev filter.....	52
Figure 2.49: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.21 with SNR = 30 dB.....	54
Figure 2.50: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 25 dB.....	54
Figure 2.51: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 20 dB.....	55

Figure 2.52: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 15 dB.....	55
Figure 2.53: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 10 dB.....	55
Figure 2.54: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 0 dB.....	56
Figure 2.55: Band-pass Chebyshev filter applied to synthetic noise trace shown in Figure 1.27.....	56
Figure 2.56: Band-pass Chebyshev filter applied to Event #1, a “good” event.....	56
Figure 2.57: Band-pass Chebyshev filter applied to Event #2, a “good” event.....	56
Figure 2.58: Band-pass Chebyshev filter applied to Event #3, a “good” event.....	57
Figure 2.59: Band-pass Chebyshev filter applied to Event #4, a “good” event.....	57
Figure 2.60: Band-pass Chebyshev filter applied to Event #5, a “good” event.....	57
Figure 2.61: Band-pass Chebyshev filter applied to Event #6, a “good” event.....	57
Figure 2.62: Band-pass Chebyshev filter applied to Event #7, a “good” event.....	58
Figure 2.63: Band-pass Chebyshev filter applied to Event #8, a “good” event.....	58
Figure 2.64: Band-pass Chebyshev filter applied to Event #9, a “good” event.....	58
Figure 2.65: Band-pass Chebyshev filter applied to Event #10, a “good” event.....	58
Figure 2.66: Band-pass Chebyshev filter applied to Event #11, a noise event.....	59
Figure 2.67: Band-pass Chebyshev filter applied to Event #12, a noise event.....	59
Figure 2.68: Band-pass Chebyshev filter applied to Event #13, a noise event.....	59
Figure 2.69: Band-pass Chebyshev filter applied to Event #14, a noise event.....	59
Figure 2.70: Band-pass Chebyshev filter applied to Event #15, a noise event.....	60
Figure 3.1: A “good” event with an empirically picked P-wave event-length of 31 ms....	66
Figure 3.2: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 2$ ms. The automatically picked event-length is 3.6 ms.....	66

Figure 3.3: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 6$ ms. The automatically picked event-length is 9.1 ms.....	66
Figure 3.4: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 10$ ms. The automatically picked event-length is 12.6 ms.....	67
Figure 3.5: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 14$ ms. The automatically picked event-length is 17.2 ms.....	67
Figure 3.6: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 20$ ms. The automatically picked event-length is 30.3 ms.....	67
Figure 3.7: Calculated STA / LTA ratios of the trace in Figure 3.1 with $S = 50$ ms. The automatically picked event-length is 52.4 ms.....	68
Figure 3.8: General block diagram of the developed continuous-time frequency-domain event-length detection algorithm. The row-vector represents input time data elements $d_i$ . The output column-vectors are $out_1, out_2$ , etc. The columns in the 2-D matrix are denoted by $C_1, C_2$ , etc.....	74
Figure 3.9: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an FFT window size $k = 10$ . There is strong time resolution, but poor frequency resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).....	75
Figure 3.10: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an FFT window size $k = 10$ . There is strong time resolution, but poor frequency resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).....	76
Figure 3.11: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an FFT window size $k = 1000$ . There is strong frequency resolution, but poor time resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).....	76
Figure 3.12: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an FFT window size $k = 1000$ . There is strong frequency resolution, but poor time resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).....	76
Figure 3.13: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an experimentally determined optimal FFT window size $k = 100$ . There is adequately strong time and frequency resolution. The event-length was calculated to be 35 ms.....	77

Figure 3.14: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an experimentally determined optimal FFT window size $k = 100$ . There is adequately strong time and frequency resolution. The event-length was calculated to be 547 ms.....	77
Figure 3.15: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.21 (SNR = 30 dB) with FFT window size $k = 100$ . The P-wave event-length was calculated to be 30 ms.....	79
Figure 3.16: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.22 (SNR = 25 dB) with FFT window size $k = 100$ . The P-wave event-length was calculated to be 30 ms.....	79
Figure 3.17: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.23 (SNR = 20 dB) with FFT window size $k = 100$ . The P-wave event-length was calculated to be 30 ms.....	79
Figure 3.18: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.24 (SNR = 15 dB) with FFT window size $k = 100$ . The P-wave event-length was calculated to be 30 ms.....	80
Figure 3.19: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.25 (SNR = 10 dB) with FFT window size $k = 100$ . The P-wave event-length was calculated to be 30 ms, but significant noise frequency components increase the difficulty of distinguishing this arrival.....	80
Figure 3.20: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.26 (SNR = 0 dB) with FFT window size $k = 100$ . The P-wave arrival is indistinguishable due to very large noise frequency components.....	80
Figure 3.21: Continuous-time Fourier transform PSD plot of synthetic noise trace in Figure 1.27 with FFT window size $k = 100$ . The calculated event-length was 1.1 seconds.....	81
Figure 3.22: Continuous-time Fourier transform PSD plot of Event #2, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 32 ms.....	81
Figure 3.23: Continuous-time Fourier transform PSD plot of Event #3, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 62 ms.....	81
Figure 3.24: Continuous-time Fourier transform PSD plot of Event #4, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 43 ms.....	82

Figure 3.25: Continuous-time Fourier transform PSD plot of Event #5, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 37 ms.....	82
Figure 3.26: Continuous-time Fourier transform PSD plot of Event #6, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 34 ms.....	82
Figure 3.27: Continuous-time Fourier transform PSD plot of Event #7, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 45 ms.....	83
Figure 3.28: Continuous-time Fourier transform PSD plot of Event #8, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 44 ms.....	83
Figure 3.29: Continuous-time Fourier transform PSD plot of Event #9, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 47 ms.....	83
Figure 3.30: Continuous-time Fourier transform PSD plot of Event #10, a “good” event, with FFT window size $k = 100$ . The calculated event-length was 54 ms.....	84
Figure 3.31: Continuous-time Fourier transform PSD plot of Event #12, a noise event, with FFT window size $k = 100$ . The calculated event-length was 114 ms.....	84
Figure 3.32: Continuous-time Fourier transform PSD plot of Event #13, a noise event, with FFT window size $k = 100$ . The calculated event-length was 234 ms.....	84
Figure 3.33: Continuous-time Fourier transform PSD plot of Event #14, a noise event, with FFT window size $k = 100$ . The calculated event-length was 570 ms.....	85
Figure 3.34: Continuous-time Fourier transform PSD plot of Event #15, a noise event, with FFT window size $k = 100$ . The calculated event-length was 415 ms.....	85
Figure 4.1: Example “good” signal.....	88
Figure 4.2: Example noise signal.....	88
Figure 4.3: Discrete time-series plot of example “good” trace for $t = 1.14\text{s}$ to $t = 1.15\text{s}$ . The dashed line represents zero amplitude to better view polarity reversals. Fewer polarity reversals between adjacent data points are seen compared to the noise trace in Figure 4.4. Additionally, the magnitudes of signed amplitude differences between adjacent time-series data points are smaller compared to the example noise trace.....	98

Figure 4.4: Discrete time-series plot of example noise trace for $t = 1.14\text{s}$ to $t = 1.15\text{s}$ . The dashed line represents zero amplitude to better view polarity reversals. More polarity reversals between adjacent data points are seen compared to the “good” trace in Figure 4.3. Additionally, the magnitudes of signed amplitude differences between adjacent time-series data points are larger compared to the example “good” trace.....	98
Figure 4.5: Histogram plot corresponding to synthetic “good” trace (SNR = 30 dB) shown in Figure 1.21.....	103
Figure 4.6: Histogram plot corresponding to synthetic “good” trace (SNR = 25 dB) shown in Figure 1.22.....	103
Figure 4.7: Histogram plot corresponding to synthetic “good” trace (SNR = 20 dB) shown in Figure 1.23.....	103
Figure 4.8: Histogram plot corresponding to synthetic “good” trace (SNR = 15 dB) shown in Figure 1.24.....	104
Figure 4.9: Histogram plot corresponding to synthetic “good” trace (SNR = 10 dB) shown in Figure 1.25.....	104
Figure 4.10: Histogram plot corresponding to synthetic “good” trace (SNR = 0 dB) shown in Figure 1.26.....	104
Figure 4.11: Histogram plot corresponding to synthetic noise event shown in Figure 1.27.....	105
Figure 4.12: Histogram plot corresponding to Event #1, a “good” trace.....	105
Figure 4.13: Histogram plot corresponding to Event #2, a “good” trace.....	105
Figure 4.14: Histogram plot corresponding to Event #3, a “good” trace.....	106
Figure 4.15: Histogram plot corresponding to Event #4, a “good” trace.....	106
Figure 4.16: Histogram plot corresponding to Event #5, a “good” trace.....	106
Figure 4.17: Histogram plot corresponding to Event #6, a “good” trace.....	106
Figure 4.18: Histogram plot corresponding to Event #7, a “good” trace.....	107
Figure 4.19: Histogram plot corresponding to Event #8, a “good” trace.....	107
Figure 4.20: Histogram plot corresponding to Event #9, a “good” trace.....	107

Figure 4.21: Histogram plot corresponding to Event #10, a “good” trace.....	107
Figure 4.22: Histogram plot corresponding to Event #11, a noise trace.....	108
Figure 4.23: Histogram plot corresponding to Event #12, a noise trace.....	108
Figure 4.24: Histogram plot corresponding to Event #13, a noise trace.....	108
Figure 4.25: Histogram plot corresponding to Event #14, a noise trace.....	108
Figure 4.26: Histogram plot corresponding to Event #15, a noise trace.....	109
Figure 5.1: Normalized algorithm output of low-pass filtering algorithm described in section 2.3. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.....	117
Figure 5.2: Normalized algorithm output of high-pass filtering algorithm described in section 2.4. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.....	117
Figure 5.3: Normalized algorithm output of band-pass filtering algorithm described in section 2.5. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.....	118
Figure 5.4: Normalized algorithm output of STA / LTA event-length detection algorithm described in section 3.2. Each pink square (blue diamond) pertains to a normalized event-length measurement for a single “good” (noise) microseismic file.....	118
Figure 5.5: Normalized algorithm output of time-localized frequency transform event-length detection algorithm described in section 3.3. Each pink square (blue diamond) pertains to a normalized event-length measurement for a single “good” (noise) microseismic file.....	119
Figure 5.6: Normalized algorithm output of Threshold statistical analysis algorithm described in section 4.3.1. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.....	119
Figure 5.7: Normalized algorithm output of Histogram statistical analysis algorithm described in section 4.3.2. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.....	120
Figure 5.8: Normalized algorithm output of Specialized Zero-Crossing Count statistical analysis algorithm described in section 4.3.3. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.....	120

Figure 5.9: Eight-dimensional dataset projected onto first principal component. Reduction in vertical overlap is seen, but significant overlap still remains.....	125
Figure 5.10: Eight-dimensional dataset projected onto second principal component. No useful information can be extracted, as only noise is seen.....	125
Figure 5.11: Eight-dimensional dataset projected onto third principal component. No useful information can be extracted, as only noise is seen.....	126
Figure 5.12: Eight-dimensional dataset projected onto fourth principal component. No useful information can be extracted, as only noise is seen.....	126
Figure 5.13: Eight-dimensional dataset projected onto fifth principal component. No useful information can be extracted, as only noise is seen.....	127
Figure 5.14: Eight-dimensional dataset projected onto sixth principal component. No useful information can be extracted, as only noise is seen.....	127
Figure 5.15: Eight-dimensional dataset projected onto seventh principal component. No useful information can be extracted, as only noise is seen.....	128
Figure 5.16: Eight-dimensional dataset projected onto eighth principal component. No useful information can be extracted, as only noise is seen.....	128
Figure 5.17: Three-dimensional dataset projected onto first principal component. No vertical overlap is seen between “good” and noise points. For this specific test dataset, perfect classification accuracy could be achieved using PCA and examining this component.....	132
Figure 5.18: Three-dimensional dataset projected onto second principal component. No useful information can be extracted, as only noise is seen.....	133
Figure 5.19: Three-dimensional dataset projected onto third principal component. No useful information can be extracted, as only noise is seen.....	133
Figure 6.1: “Event_Analyzer” program upon startup.....	137
Figure C.1: Sketches of hypothetical crossplots to illustrate significance of covariance calculations between two example variables $B_1$ and $B_2$ . a) Negligible covariance. b) Large positive covariance. c) Large negative covariance.....	159
Figure C.2: Illustrative 2-D crossplot example of measurements taken for two variables with zero-mean correction applied. The first variable is plotted with respect to the $x$ axis, and the second is plotted with respect to the $y$ axis. Strong positive correlation is seen between the two variables.....	168

Figure C.3: Illustrative 2-D crossplot example from Figure C.2 with principal component vectors  $\hat{\mathbf{E}}_1$  and  $\hat{\mathbf{E}}_2$  shown. Maximum data variance is seen in the direction of  $\hat{\mathbf{E}}_1$ , while mainly noise is seen in the direction of  $\hat{\mathbf{E}}_2$ . Thus, data projected onto  $\hat{\mathbf{E}}_1$  would yield virtually all important information about the dataset, resulting in the effective dimensionality of the data reduced from two to one.....168

## LIST OF SYMBOLS AND ABBREVIATIONS

(Page 1 of 3)

Symbol	Definition
API	American Petroleum Institute
EOR	enhanced oil recovery
CSS	cyclic steam stimulation
DC	direct current
PCA	principal components analysis
$\omega$	angular frequency
$\omega_p$	passband angular frequency limit
$\omega_s$	stopband angular frequency limit
$\Omega$	normalized angular frequency
$\alpha(\omega)$	filter attenuation function
$\alpha_{min}$	minimum allowable stopband attenuation
$\alpha_{max}$	maximum allowable passband attenuation
$T(j\omega)$	filter transfer function
$C_n(\Omega)$	Inverse-Chebyshev function
$\varepsilon$	filter tolerance
$d_i$	data point element

## LIST OF SYMBOLS AND ABBREVIATIONS

(Page 2 of 3)

Symbol	Definition
STA	short-term energy average
LTA	long-term energy average
$R$	STA / LTA ratio
$N$	total number of data samples
$t$	time
PSD	power spectral density
DFT	discrete Fourier transform
$f_s$	sampling frequency
$f_{NYQ}$	Nyquist frequency
FFT	fast Fourier transform
$W$	an FFT window
$\mu$	rigidity, or mean
$\sigma^2$	variance
<b>E[]</b>	expected value
$\rho_i$	occurrence probability
<b>VAR[]</b>	dataset variance
$b$	histogram bin width

## LIST OF SYMBOLS AND ABBREVIATIONS

(Page 3 of 3)

Symbol	Definition
$Q_{13}$	dataset interquartile range
<b>A</b>	normalized observation matrix
<b>B</b>	mean-corrected normalized observation matrix
<b>C</b>	covariance matrix of <b>B</b>
<b>V</b>	principal component matrix
<b>P</b>	the matrix <b>B</b> projected onto principal components
<b>D</b>	covariance matrix of <b>P</b>
$\mathbf{E}_i$	eigenvector
$\hat{\mathbf{E}}_i$	unit-length eigenvector
$\lambda_i$	eigenvalue
<b>I</b>	identity matrix
<b>F</b>	matrix with unit-length eigenvectors in columns
<b>H</b>	arbitrary diagonal matrix

## **CHAPTER ONE: Introduction**

### **1.1 Passive-Seismic Monitoring**

Passive-seismic monitoring listens for small earthquakes (microseisms) that can occur when there are stress changes in a reservoir (Maxwell and Urbanic, 2001). As opposed to conventional surface-seismic acquisition techniques, passive-seismic monitoring does not employ a source to create elastic waves. Instead, spontaneous subsurface microseismic events are detected with sensors when they occur.

In oil fields, passive-seismic monitoring has various applications (Maxwell and Urbanic, 2001). When hydraulic fracturing occurs, microseismic signal characteristics can be analyzed to create images that may lead to inferences of the fractures' location, size, orientation, complexity, and temporal growth. Microseismicity is also used in deformation monitoring of rock masses to detect and locate casing failures in producing wells. Other applications of microseismic monitoring include the mapping of fluid injection, fluid movements, compaction strains, and thermal fronts. Passive-seismic monitoring can also be used for structural mapping (Duncan, 2005).

Various exploration and production companies are currently involved in microseismic monitoring. Some examples include Shell Canada and Imperial Oil at the Peace River and Cold Lake heavy oil fields, respectively, in Alberta, Canada; and Phillips Petroleum and Shell International Exploration and Production at the Ekofisk and Valhall offshore oil fields, respectively, in the Norwegian sector of the North Sea.

## 1.2 Cold Lake Oil Sands

### 1.2.1 Background

In the eastern portion of the Western Canadian Sedimentary Basin, there is a trend of heavy oil accumulations that were deposited during the early Cretaceous geologic period (Isaac, 1996). The oil sands deposit at Cold Lake, Alberta, is located in the east-central part of the province, and is one of three such deposits in Alberta, as shown in Figure 1.1 (Imperial Oil Ltd., 2006a). The Cold Lake oil sands encompass an area of approximately 6,500 km<sup>2</sup> from Townships 52 to 67 north, and Ranges 4 to 10 west of the fourth meridian (Isaac, 1996).



Figure 1.1: Alberta's three major oil sands deposits. The Cold Lake deposit is located in the east-central part of the province (Imperial Oil Ltd., 2006a).

### 1.2.2 Imperial Oil Operations at Cold Lake

Imperial Oil Limited is involved in oil sands production at Cold Lake, where there are currently approximately 850 million barrels of proved reserves, 3200 operating wells, and over 120,000 barrels of bitumen produced each day. Figure 1.2 (Imperial Oil Ltd., 2006b) shows the location of Imperial's Cold Lake operations.

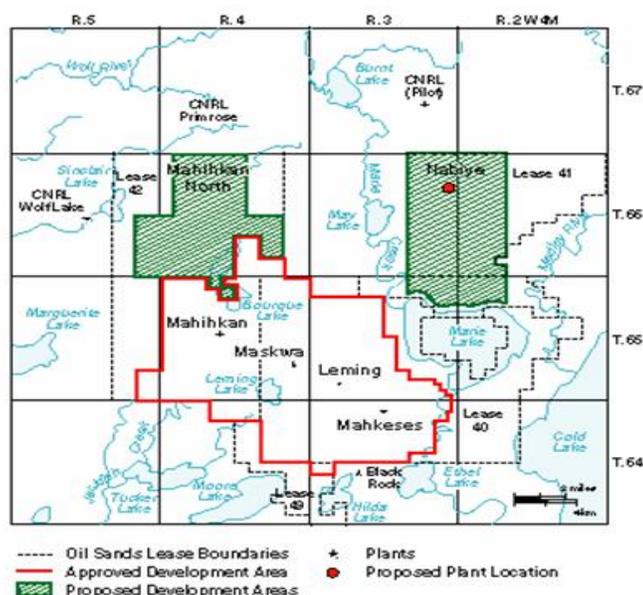


Figure 1.2: Imperial's Cold Lake operations (Imperial Oil Ltd., 2006b).

A simplified stratigraphic chart of the Cold Lake area is shown in Figure 1.3 (Isaac, 1996). Hydrocarbon production comes from the Clearwater formation, which has a primarily sandstone lithology. This producing formation is buried over 400 m deep at Cold Lake, and the bitumen contained within it has an American Petroleum Institute (API) gravity index of approximately 8° to 9°. Thus, surface mining techniques are not possible, and an enhanced oil recovery (EOR) technique is required. The EOR used by Imperial at Cold Lake is cyclic steam stimulation (CSS), sometimes also referred to as the “huff-and-puff” method. During CSS, production wells endure cycles of steam injection, soaking, and

production. Steam injection with durations of weeks to months is the first phase in the CSS cycle. The soak phase which follows allows the injected heat to soak into the producing formation and reduce the bitumen viscosity. This phase typically lasts from days to weeks. When the bitumen viscosity is sufficiently lowered for production, hot oil is pumped to the surface until the production rate declines past a predetermined threshold, giving way to another injection-soak-production cycle. This cycle is repeated so long as it is economically viable. Figure 1.4 (Imperial Oil Ltd., 2006c) depicts the CSS process for a single production well.

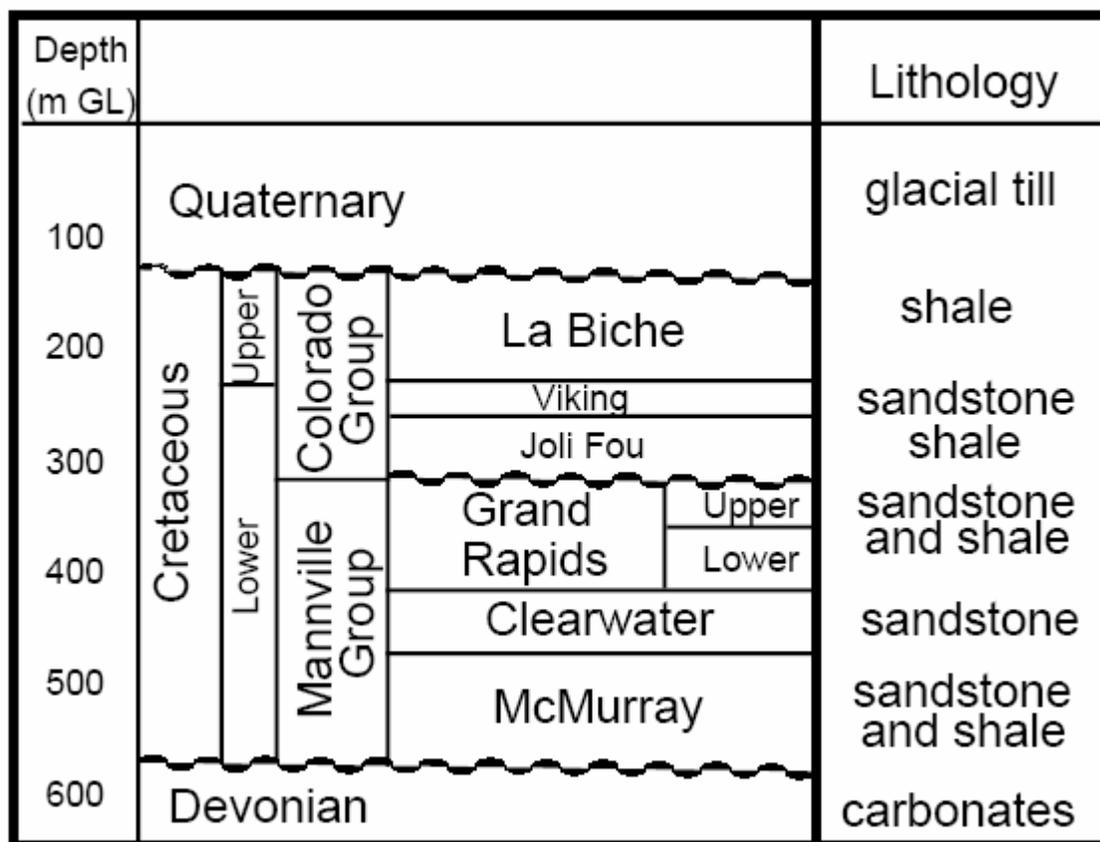


Figure 1.3: Simplified stratigraphic chart of Cold Lake area (Isaac, 1996).

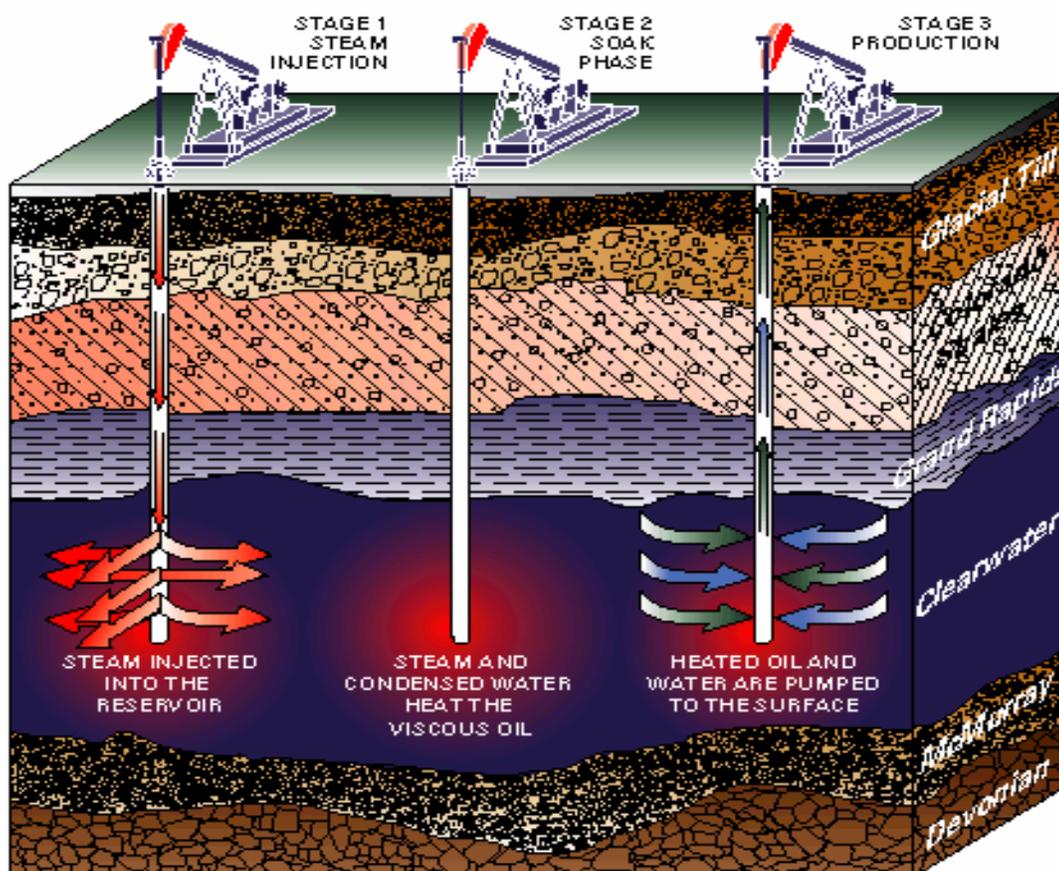


Figure 1.4: CSS enhanced oil recovery process for a single production well. The well is put through cycles of steam injection, soaking, and production, which is repeated so long as it is economically viable (Imperial Oil Ltd., 2006c).

### 1.2.3 Passive-Seismic Monitoring Requirement

There are pertinent issues associated with the CSS process as applied to the oil sands at Cold Lake. When using CSS for recovery at Cold Lake, pressures and temperatures of approximately 320°C and 11 MPa, respectively, are created in the Clearwater formation (Campbell, 2005). Mechanical issues in the producing wells such as cement cracks or casing failures can result from these high pressures and temperatures. If undetected, these production issues could result in large cleanup costs, in addition to

potential legal implications. For example, a casing failure could potentially cause environmental damage such as aquifer contamination. A microseismic earthquake with its focus near the damaged area is created when these mechanical issues occur. Imperial Oil Limited operates a passive-seismic monitoring system at Cold Lake to proactively detect these microseisms so that prompt action can be taken if a production issue is detected. The CREWES Project at the University of Calgary was involved in this passive-seismic research with Imperial Oil.

The passive-seismic monitoring system implemented at Cold Lake is present on approximately 75 production pads, each of which contain about 18 to 24 producing wells (Campbell, 2005). Each pad has a centrally located monitoring well that records ground vibrations (including microseisms). The monitoring well is instrumented by a down-hole array of five or eight 3-component (3-C) geophone sondes connected to seismic recorders at the surface (Tan et al., 2006). Seismic recorders listen for discrete seismic events and store them as microseismic event files to disk for later review. For an array of five (eight) geophones, these digital event files contain fifteen (twenty-four) traces that display 1.365 seconds (1.5 seconds) of microseismic activity recorded by the 3-component geophone sondes. Three traces are outputted from each 3-C geophone. Figure 1.5 shows an example schematic of a five-geophone array lowered into a monitoring well to detect microseismic events occurring in production wells. The approximate sensor depth locations shown are based on measurements by Talebi et al. (1998) pertaining to a project where they observed seismicity at Cold Lake. For this project, Talebi et al. (1998) observed received signals from five 3-C geophones spaced equally from the top of the Colorado Shales to the top of

the Clearwater Formation. The 3-C geophones were present at depths of approximately 160 m, 220 m, 280 m, 340 m, and 400 m.

Assuming the configuration shown in Figure 1.5, an example outline of a microseismic file is depicted in Figure 1.6. Since the schematic in Figure 1.5 depicts an array of five 3-C geophones, the corresponding microseismic file shown in Figure 1.6 would contain fifteen traces of microseismic activity, as each 3-C geophone outputs three traces, each trace corresponding to a single component ( $x$ ,  $y$ , or  $z$ ). For this example, each trace, represented by an arrow in Figure 1.6, contains 1.365 seconds of microseismic activity. The three traces at the top of a microseismic event file come from the shallowest geophone; the next three traces below are from the second-shallowest geophone, and so on.

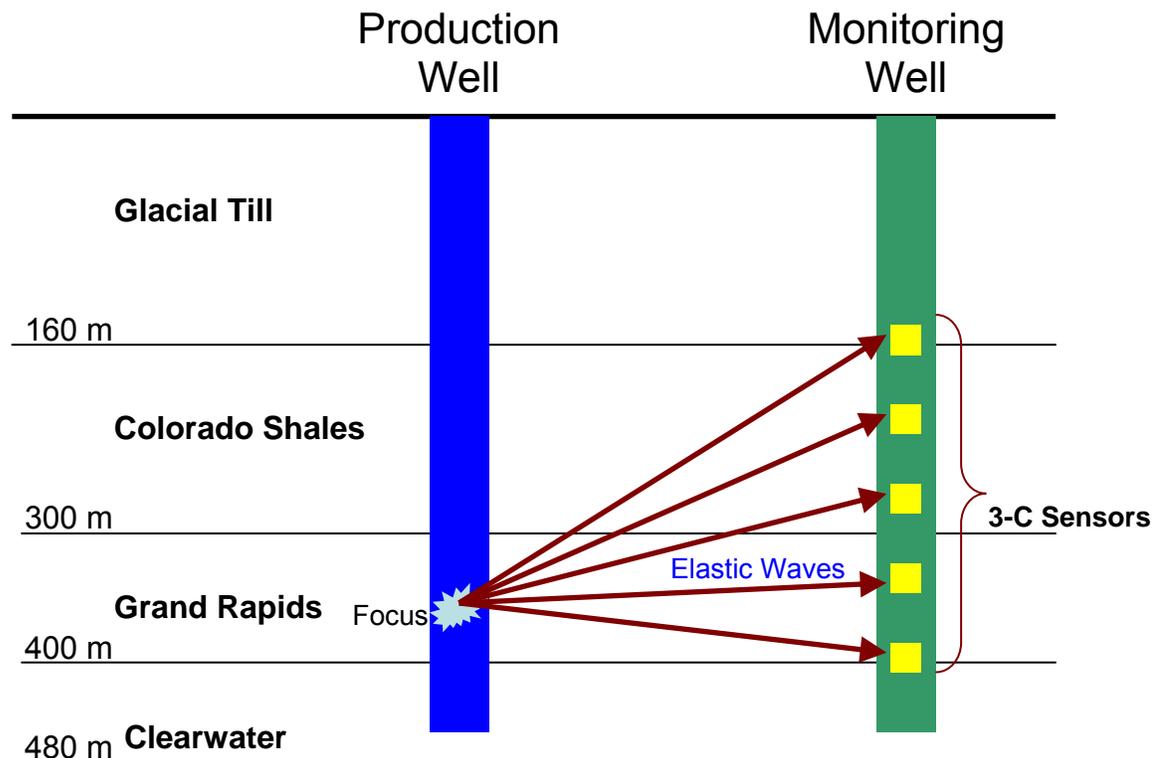


Figure 1.5: Example schematic of a five-geophone array lowered into a monitoring well to detect microseismic events occurring in production wells. When a microseism occurs, elastic P- and S-waves are propagated from its focus. The focus in this example occurs in the Grand Rapids formation, above the producing Clearwater formation.

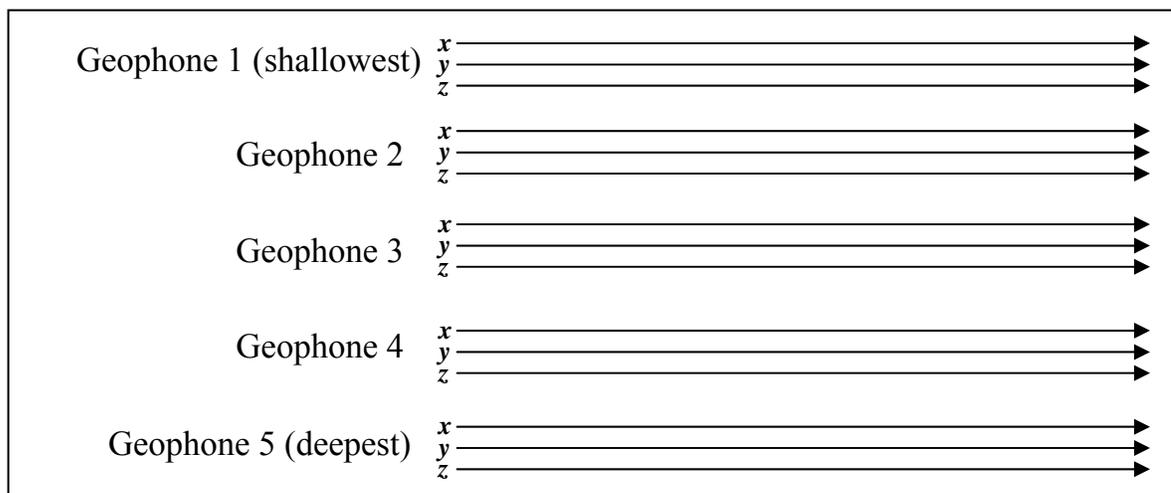


Figure 1.6: Example outline of a microseismic file assuming geophone configuration depicted in Figure 1.5. As each 3-C geophone outputs three traces, one for each component ( $x$ ,  $y$ , and  $z$ ), this example file contains fifteen traces of microseismic activity. Each trace in this example has a temporal duration of 1.365 seconds. The three traces at the top of a microseismic event file come from the shallowest geophone; the next three traces below are from the second-shallowest geophone, and so on. Each trace is represented by an arrow in this figure.

The 3-component geophone sensors deployed in monitoring wells require strong coupling with surrounding rock and a wideband frequency response (Maxwell and Urbanic, 2001). Strong coupling is required between sensors and surrounding rock if microseismic vibration energy is to be efficiently transferred from the rock to the sensors. At Cold Lake, strong coupling is achieved through cementation of the sensors to surrounding rock mass (Talebi et al., 1998). A wideband sensor frequency response is required to accurately depict detected microseisms. Ideally, a sensor frequency response with infinite bandwidth (which is impossible to achieve in practice) would be desired so that sensors can record all frequency components in a detected microseism without any attenuation. In practice, a wideband sensor frequency response is targeted. The passive-seismic monitoring system at Cold Lake has been observed to have a flat frequency response for frequencies up to

1500 Hz (Talebi et al., 1998). This signifies that frequencies up to 1500 Hz are accurately detected with negligible attenuation.

Vendor-supplied event-classification software analyzes each created microseismic file and assigns a classification. If a file is classified as "good", this indicates that the software has decided that the event file warrants further investigation; conversely, if a file is classified as "noise", it is supposedly an event that is not of interest (Tan et al., 2006). Approximately 99% of all detected events are noise. Examples of "good" events worth further investigation include cement cracks around the casing in the wells, and casing failures. Examples of noise events include noise created by pump rods and passing vehicles (Campbell, 2005). Noise events are usually discarded.

#### 1.2.3.1 Microseismic Events

The characteristics of “good” events are similar to those of a small earthquake that emanates P- and S-waves from its focus (e.g. Maxwell and Urbanic, 2001). Thus, conventional earthquake seismology can be used to explain common characteristics of “good” events, as these microseismic events are essentially very small earthquakes, which are created from radiated energy that is released when rock mass deformations rebound in the producing formation’s overburden due to the high temperatures and pressures present during CSS. These rock mass deformations due to pressure changes in the producing reservoir cause applied stress on the overburden resulting in strain.

Theoretical earthquake seismology discussed by Scholz (1990); Lay and Wallace (1995); and Aki and Richards (2002) are outlined below to explain “good” event characteristics. Earthquakes occur when stress that is built up due to deformation of the

Earth's crust is suddenly released. This release results in energy radiated from the earthquake's source in the form of seismic elastic waves, which include P- (primary) and S- (secondary) body waves. It is estimated that less than 10 percent of the energy released by an earthquake is in the form of these seismic elastic waves. The majority of the released energy is in the form of heat. Possible causes of large earthquakes include volcanic eruptions and fault slippage. The "focus" is the location within the Earth where the P- and S-waves originate. This is the location where the sudden release of built-up elastic energy occurred. The "epicenter" is the point on the Earth's surface directly above the Earthquake's focus.

P-waves travel through material through compression, resulting in material deformation in a direction parallel to propagation. P-wave velocity depends on elastic rock properties, namely the incompressibility, rigidity, and density of the rock through which the waves travel. Incompressibility pertains to the material's resistance to volumetric dilation / compression, while rigidity corresponds to the material's resistance to shear deformation. It has been shown that

$$V_p = \sqrt{\frac{k + \frac{4}{3}\mu}{\rho}}, \quad (1.1)$$

where  $V_p$  represents P-wave velocity,  $k$  is incompressibility,  $\mu$  is rigidity, and  $\rho$  is medium density.

S-waves travel through media through shear, resulting in material deformation in a direction perpendicular to propagation. S-wave velocity depends only on rock rigidity and density. These waves travel slower than P-waves, and cannot travel through liquids, as liquids have no resistance to shearing. It has been shown that

$$V_s = \sqrt{\frac{\mu}{\rho}}, \quad (1.2)$$

where  $V_s$  represents S-wave velocity.

Many microseismic events tend to have impulsive arrivals, high P-wave arrival frequencies, and decreasing signal frequency with increasing time (Lee and Stewart, 1981). S-wave arrivals in microseismic events are expected to have lower frequencies than P-wave arrivals because, for a fixed set of source dimensions (related to source wavelength), S-waves have lower velocities compared to P-waves. Through inspection of the equation

$$f = \frac{v}{\lambda}, \quad (1.3)$$

it can be seen that if the source wavelength  $\lambda$  is held constant, then a slower velocity of propagation  $v$ , as is the case with S-waves compared to P-waves, should result in a lower signal frequency  $f$ . At Cold Lake, sonic well-logs have measured P-wave velocities in the range of 2200 m/s to 2400 m/s (Eastwood, 1993). Measured S-wave velocities are in the range of 1100 m/s to 1250 m/s. Thus, if equation 1.3 is applied, one could expect detected S-wave frequencies that are approximately half that of P-waves. A lower quality factor pertaining to shear waves has also been proposed as an explanation as to why S-wave arrivals in microseisms have lower frequencies than P-wave arrivals (e.g. DeNoyer, 1966). Microseism time-series are also non-stationary, signifying that the shape of microseismic wavelets varies with time (Correig and Urquizú, 2002).

Microseisms also tend to have higher frequencies than larger earthquakes because the dimensions of microseismic source regions are generally much smaller than those of source regions pertaining to larger earthquakes (DeNoyer, 1966). For example, many larger earthquakes are produced due to fault slippage, which takes place on a much larger spatial

scale than microseism production due to overburden deformation in a producing reservoir. As a result, seismic waves with shorter wavelengths and higher frequencies tend to be generated from microseismic sources. For example, the fault area of a conventional earthquake that generates surface waves of magnitude  $M_S = 6$  can be approximately  $1.3 \times 10^8 \text{ m}^2$  (Lay and Wallace, 1995). Surface waves, different from P- and S-waves, travel along trajectories approximately parallel to the Earth's surface, as opposed to traveling through the Earth's subsurface as is the case with P- and S- body waves. The surface wave earthquake magnitude  $M_S$  represents a relative logarithmic amplitude measurement. To compare, the fault area in a microseism of magnitude  $M_S = -2$  can be approximately  $0.9 \text{ m}^2$ .

The earthquake's "radiation pattern" affects the shape of detected waveforms. This pattern pertains to the shape of the wavefronts of emanated P- and S-waves close to the earthquake's focus (Lay and Wallace, 1995). By observing the shapes of the P- and S- arrivals, inferences of the corresponding radiation pattern can be made followed by inferences of the faulting process that caused the earthquake. Related to this is the "fault plane solution", where arriving P- and S- waveforms can be used to estimate the direction of faulting, which is parallel to the fault plane (e.g. Warren and Shearer, 2006). By examining these waveforms, the length of the fault, in addition to the rupture speed can also be estimated. This can be applied to both conventional earthquake and microseism analysis. For example, a fast rupture typically results in a radiation pattern that leads to microseisms that have larger peak amplitudes, as a greater energy burst is released compared to the case of a slow rupture. Additionally, larger fault plane areas result in radiation patterns that lead to detected waveforms with longer wavelengths and lower frequencies, while radiation

patterns pertaining to smaller fault plane areas result in waveforms with shorter wavelengths and higher frequencies (e.g. DeNoyer, 1966).

When a “good” microseismic event is automatically detected followed by manual confirmation, an attempt is made to locate its hypocenter, its point of origin. The distance from a sensor to the event’s hypocenter can be calculated by determining the time delay between the P- and S-wave arrivals. These arrival times can be picked through determining temporal locations where there are sharp increases in short-term to long-term averages of signal energy. Estimations of P- and S-wave velocities are also required to determine this distance. The orientation of the event’s focus relative to the sensors can be determined through observation of the received signals pertaining to the three detected directional components. In cases where many “good” events are detected over short time spans, it could be beneficial to plot potential event hypocenters on a map to determine where these hypocenters are most likely to be located (e.g. Campbell, 2005). Once hypocenter location has been performed, the cause of the recorded microseism is investigated further, and, if required, mechanical repairs could take place.

Figures 1.7 to 1.16 show ten examples of “good” events from the Cold Lake dataset, with P- and S-wave arrivals indicated. Each trace depicts detected motion pertaining to a single component of a 3-C geophone. Empirically, it can be determined that these events are “good” due to the distinct and impulsive P- and S-wave arrivals. As expected, the P-wave arrivals occur before the S-waves due to their higher velocity of propagation. A decrease in frequency can be seen in the S-wave arrivals compared to the P-waves in these events. Events are labeled 1 to 10 from Figure 1.7 to Figure 1.16, respectively, for future

reference when applying and demonstrating robustness of the developed classification algorithms discussed in chapters 2 to 4.

Figures 1.7 to 1.16 provide perspective on how attributes can change from one event to the next in the Cold Lake microseismic dataset. The relative amplitudes between P- and S-wave arrivals; the degree of superimposed noise; and event frequencies are characteristics of interest. As previously discussed, higher event frequencies correspond to radiation patterns pertaining to smaller fault plane areas.

Event #1 shown in Figure 1.7 is a “good” event with minimal superimposed noise with comparable P- and S-wave amplitudes. Event #2 in Figure 1.8 shows a “good” event with high-frequency noise superimposed. The P-wave arrival is still easily distinguishable and is followed by a significantly larger S-wave arrival. Event #3 in Figure 1.9 shows an event where a very impulsive, high-amplitude P-wave arrival is followed by a lower frequency, low-amplitude S-wave arrival. The exact cause for the significant decrease in shear-wave amplitude is not known, but hypotheses can be made. This amplitude decrease could be due to fluid content between the event hypocenter and the sensors, as shear waves can not travel through fluids. Also, differences between subsurface compressional and shear quality factors can cause differences between received P- and S-wave amplitudes (e.g. DeNoyer, 1966).

Event #4 in Figure 1.10 shows an event with an impulsive P-wave arrival followed by an impulsive S-wave arrival of higher amplitude. Low to moderate frequency noise is seen in this event; however, the P- and S-wave arrivals are still distinct and easily distinguishable. Event #5 in Figure 1.11 depicts an event with an impulsive P-wave arrival followed by an S-wave arrival with lower amplitude and lower frequency. Noise is

negligible in this trace. Event #6 in Figure 1.12 shows a P-wave arrival followed by a much larger S-wave arrival. Superimposed low-frequency noise is seen. Event #7 in Figure 1.13 is an event with comparable P- and S-wave arrival amplitudes. Noise occurs between the P- and S-wave arrivals.

Event #8 in Figure 1.14 shows an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and low-frequency noise. Event #9 in Figure 1.15 shows an impulsive P-wave arrival followed by noise with wideband frequency components, an S-wave arrival of smaller amplitude, and low-frequency noise. Event #10 in Figure 1.16 shows an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and negligible noise thereafter.

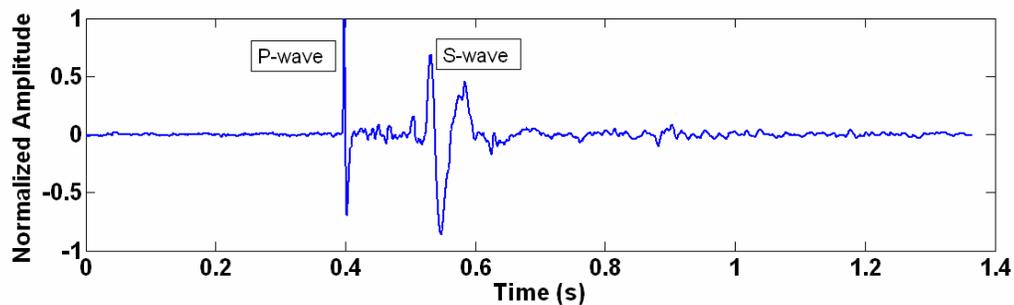


Figure 1.7: Example trace of a “good” event with minimal superimposed noise. The impulsive P- and S-wave arrival signatures indicate the presence of a “good” event. This “good” event is labeled “Event #1” for future reference.

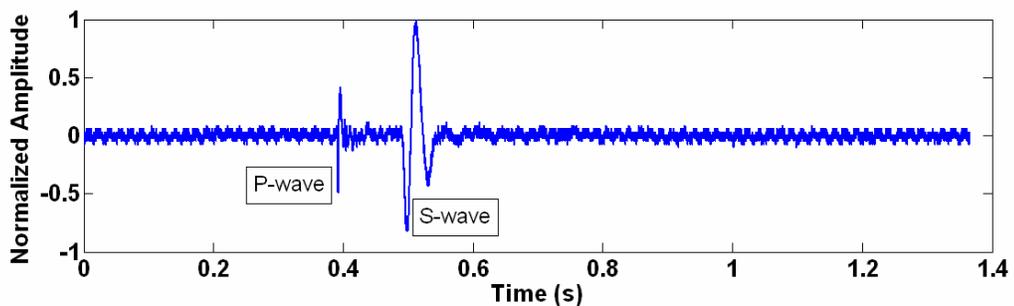


Figure 1.8: A “good” event with high-frequency noise superimposed. The P-wave arrival is still easily distinguishable and is followed by a significantly larger S-wave arrival. This “good” event is labeled “Event #2” for future reference.

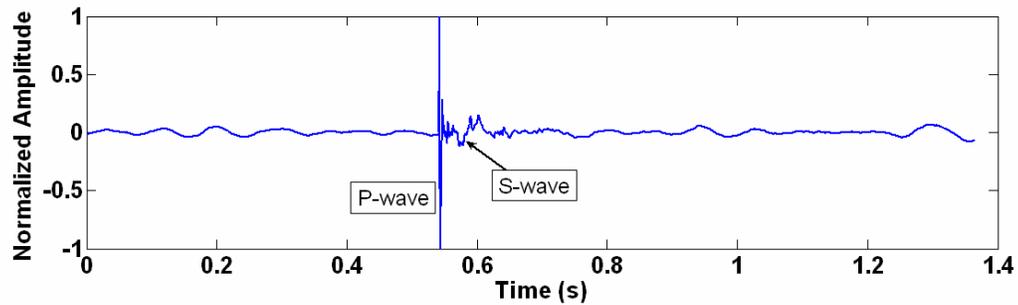


Figure 1.9: A “good” event where a very impulsive, high-amplitude P-wave arrival is followed by a lower frequency, low-amplitude S-wave arrival. This amplitude decrease could be due to fluid content between the event hypocenter and the sensors and differences between subsurface compressional and shear quality factors. This “good” event is labeled “Event #3” for future reference.

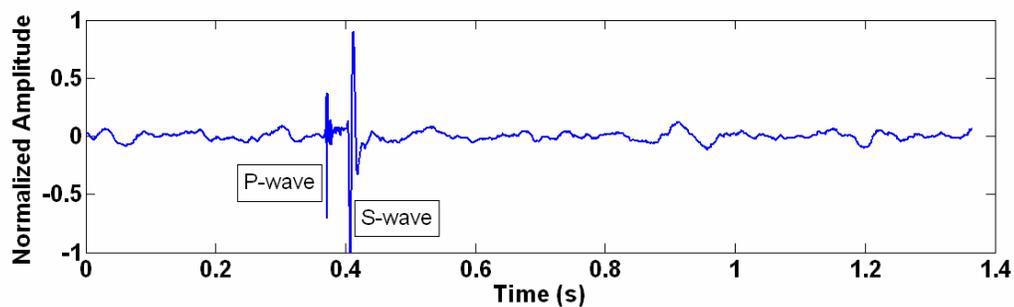


Figure 1.10: A “good” event with an impulsive P-wave arrival followed by an impulsive S-wave arrival of higher amplitude. Low to moderate frequency noise is seen in this event, but P- and S-wave arrivals are still distinct and easily distinguishable. This “good” event is labeled Event #4 for future reference.

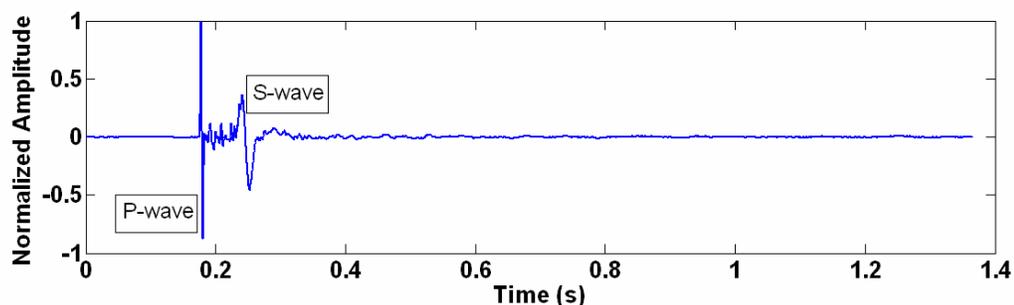


Figure 1.11: A “good” event with an impulsive P-wave arrival followed by an S-wave arrival with lower amplitude and lower frequency. Noise is negligible in this trace. This “good” event is labeled Event #5 for future reference.

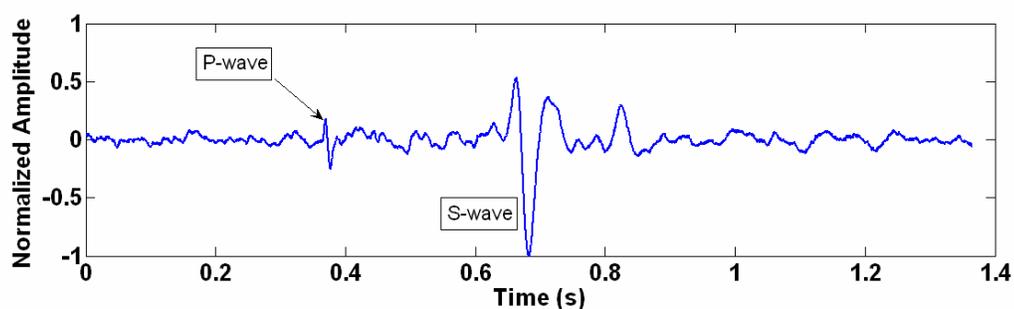


Figure 1.12: A “good” event with a P-wave arrival followed by a much larger S-wave arrival. Superimposed low-frequency noise is seen. This “good” event is labeled Event #6 for future reference.

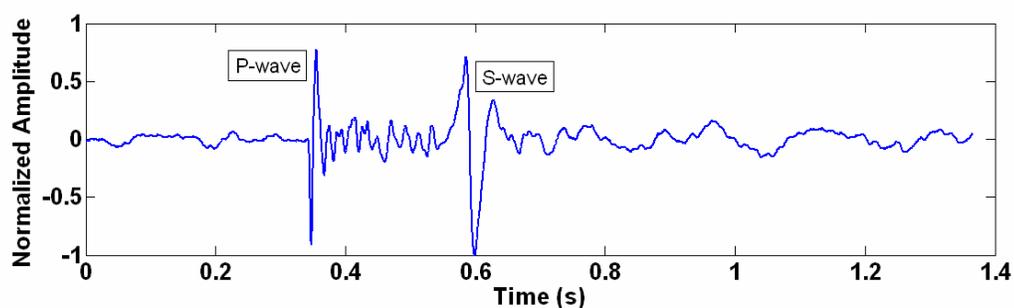


Figure 1.13: A “good” event with comparable P- and S-wave arrival amplitudes. Noise occurs between the P- and S-wave arrivals. This “good” event is labeled Event #7 for future reference.

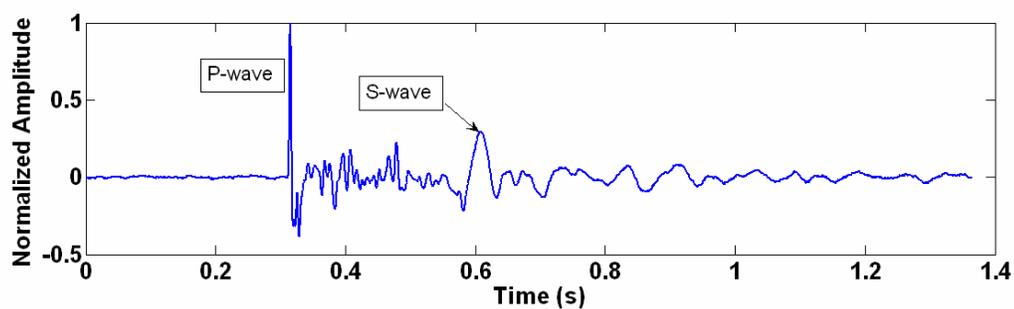


Figure 1.14: A “good” event with an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and low-frequency noise. This “good” event is labeled Event #8 for future reference.

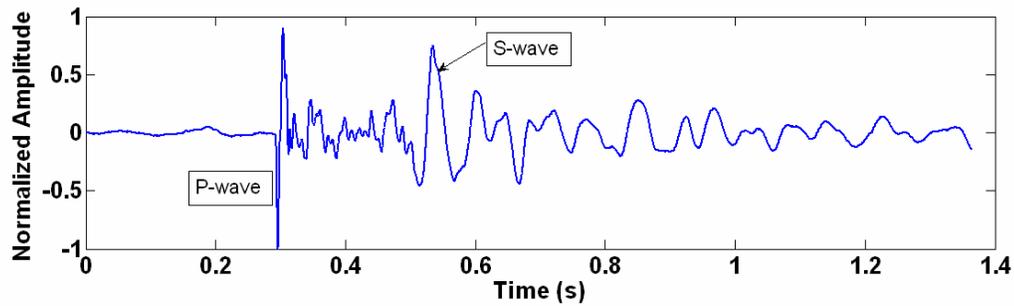


Figure 1.15: A “good” event with an impulsive P-wave arrival followed by noise with wideband frequency components, an S-wave arrival of smaller amplitude, and low-frequency noise. This “good” event is labeled Event #9 for future reference.

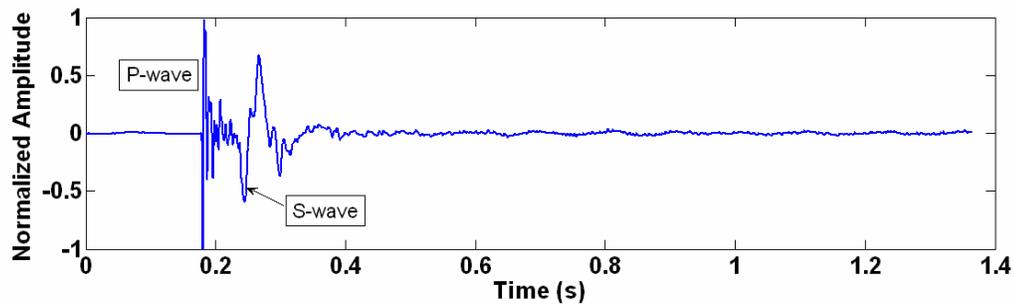


Figure 1.16: A “good” event with an impulsive P-wave arrival followed by high-frequency noise, an S-wave arrival of lower amplitude, and negligible noise thereafter. This “good” event is labeled Event #10 for future reference.

Figures 1.17 to 1.21 show five examples of noise events. As noise events can be caused in a wide variety of ways including surface noise due to passing vehicles and pump rods operating in producing wells, noise events tend to be random and do not follow the discussed deterministic properties associated with “good” events. The noise events in Figure 1.17 to Figure 1.21 are labeled as Events 11 to 15, respectively, for future reference when applying and demonstrating robustness of the developed classification algorithms discussed in chapters 2 to 4.

Figures 1.17, 1.18, and 1.19 contain events of relatively long duration, which could be caused by low-frequency (long duration) pump rod oscillation in production wells.

Figure 1.19 shows a noise trace with two such events. Figure 1.20 shows a noise trace containing a relatively impulsive, high frequency noise event, possibly caused by high-frequency (short duration) pump rod oscillation in production wells. Figure 1.21 shows random, lower frequency noise. This is likely random jitter detected by the sensors, as noise due to oscillating pump rods and passing vehicles tend to cause microseisms of higher frequencies.

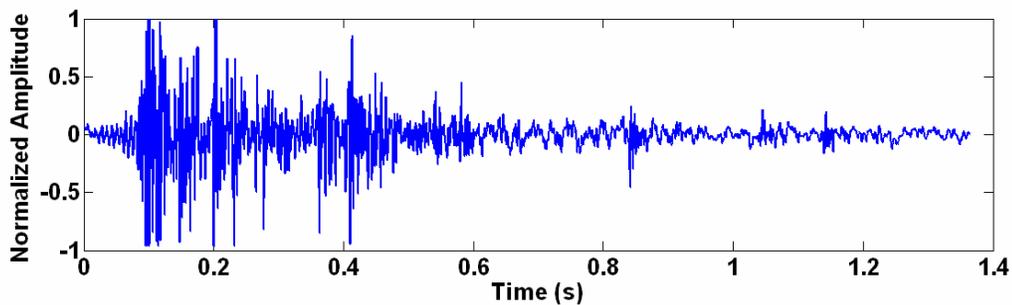


Figure 1.17: Example trace of a noise event. This event could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #11 for future reference.

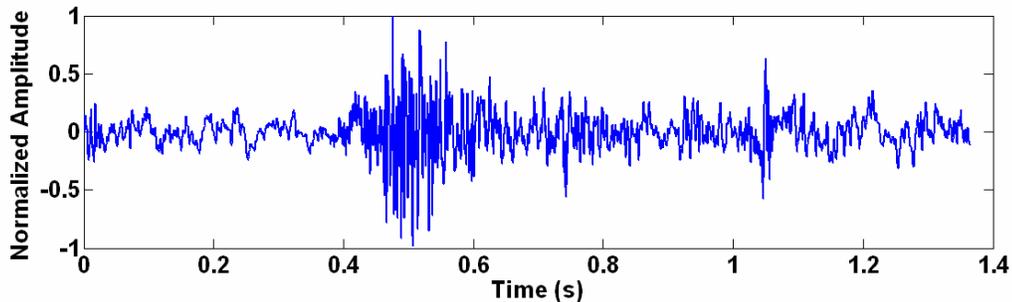


Figure 1.18: A noise trace containing an event of long duration. This event could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #12 for future reference.

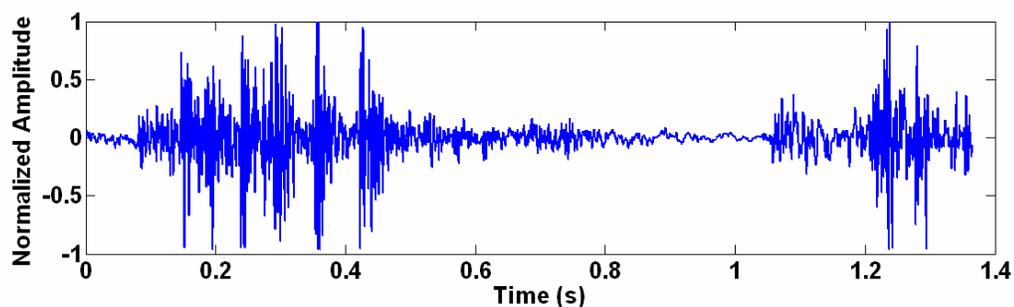


Figure 1.19: A noise trace containing two events of long duration. These events could be caused by low-frequency (long duration) pump rod oscillation in production wells. This noise event is labeled Event #13 for future reference.

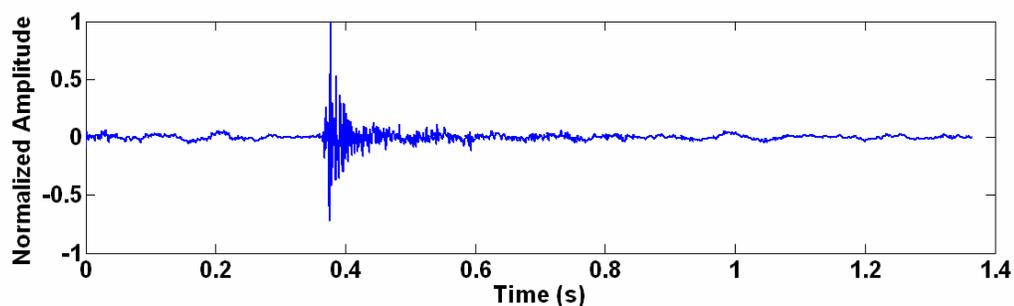


Figure 1.20: A noise trace containing a relatively impulsive, high frequency noise event, possibly caused by high-frequency (short duration) pump rod oscillation in production wells. This noise event is labeled Event #14 for future reference.

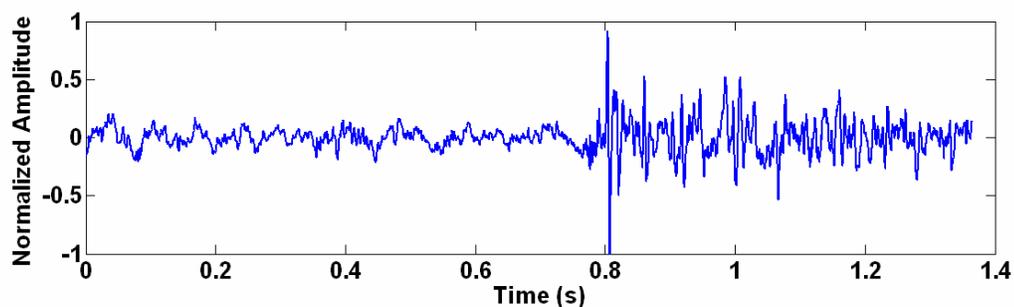


Figure 1.21: Random, lower frequency noise. This is likely random jitter detected by the sensors, as noise due to oscillating pump rods and passing vehicles tend to cause microseisms of higher frequencies. This noise event is labeled Event #15 for future reference.

The example traces shown in Figures 1.7 to 1.21, in addition to the developed synthetic data in section 1.3, will be used when demonstrating the classification algorithms. These traces are normalized to the largest data value (in magnitude), and have any DC offset removed.

### 1.3 Synthetic Data

Synthetic events pertaining to 1.365 seconds of microseismic activity were created as an additional test for algorithms developed to address the microseismic file-classification issue described in section 1.4. Synthetics representing “good” events with varying signal-to-noise ratios were developed to test algorithm robustness. A synthetic noise event was also created. Synthetic parameters such as P- and S-wave event-lengths and frequencies were chosen to reasonably represent observed characteristics in the Cold Lake microseismic dataset.

#### 1.3.1 “Good” Events

Synthetic “good” events containing impulsive, single-cycle P- and S-wave arrivals with varying signal-to-noise ratios were created.

Define a sampling frequency  $f_s$  to be  $f_s = 3000$  Hz. The sample time interval is then  $\Delta t = \frac{1}{f_s}$ . Let  $s_1(t)$  represent an impulsive P-wave arrival with frequency  $f_p = 100$  Hz resulting in an angular frequency of  $\omega_p = 2\pi f_p$  and a period of  $T_p = \frac{1}{f_p}$ . Define  $t_p$  as the

time delay before the P-wave arrival and  $A_s$  as a signal amplitude to be determined.

Then  $s_1(t)$  can be expressed as

$$\begin{aligned} s_1(t) &= A_s \sin(\omega_p(t - t_p)), & t_p \leq t \leq t_p + T_p; \\ s_1(t) &= 0, & \text{otherwise.} \end{aligned} \quad (1.4)$$

Similarly, let  $s_2(t)$  represent a S-wave arrival with frequency  $f_s = 50$  Hz resulting in an angular frequency of  $\omega_s = 2\pi f_s$  and a period of  $T_s = \frac{1}{f_s}$ . Define  $t_s$  as the time delay

before the S-wave arrival. Then  $s_2(t)$  can be expressed as

$$\begin{aligned} s_2(t) &= A_s \sin(\omega_s(t - t_s)), & t_s \leq t \leq t_s + T_s; \\ s_2(t) &= 0, & \text{otherwise.} \end{aligned} \quad (1.5)$$

Define  $s_T(t)$  as a composite signal containing both the P- and S-wave arrivals given by

$$s_T(t) = s_1(t) + s_2(t). \quad (1.6)$$

A noise component will be superimposed onto  $s_T(t)$ . Let  $n(t)$  represent this noise component given by

$$n(t) = N, \quad (1.7)$$

where  $N$  is a uniformly-distributed random variable in the range  $-A_n \leq N \leq A_n$ . This signifies that for any discrete time sample, it is equally probable for the random variable  $N$  to obtain any values between  $-A_n$  and  $A_n$ . A “good” event,  $s_g(t)$ , can now be created by combining the developed signal and noise signals as

$$s_g(t) = s_T(t) + n(t). \quad (1.8)$$

The relationship between the signal and noise amplitudes  $A_s$  and  $A_n$ , respectively, is determined by deciding upon a signal-to-noise ratio (SNR), which is defined as the ratio of

signal power to noise power. Since power is the square of amplitude, SNR can be defined as

$$\text{SNR} = \frac{\text{signal power}}{\text{noise power}} = \frac{(\text{signal amplitude})^2}{(\text{noise amplitude})^2} = \frac{A_s^2}{A_n^2}. \quad (1.9)$$

In decibels, the signal-to-noise ratio is expressed as

$$\text{SNR}_{\text{dB}} = 10 \log(\text{SNR}) = 20 \log\left(\frac{A_s}{A_n}\right) \text{ dB}. \quad (1.10)$$

After choosing an SNR value to determine the relationship between  $A_s$  and  $A_n$ , the “good” event  $s_g(t)$  is static shifted to have zero mean and is normalized to an amplitude range from -1 to 1.

Synthetic “good” events were created pertaining to SNR values of 30 dB, 25 dB, 20 dB, 15 dB, 10 dB, and 0 dB. These events are shown in Figures 1.22, 1.23, 1.24, 1.25, 1.26, and 1.27, respectively.

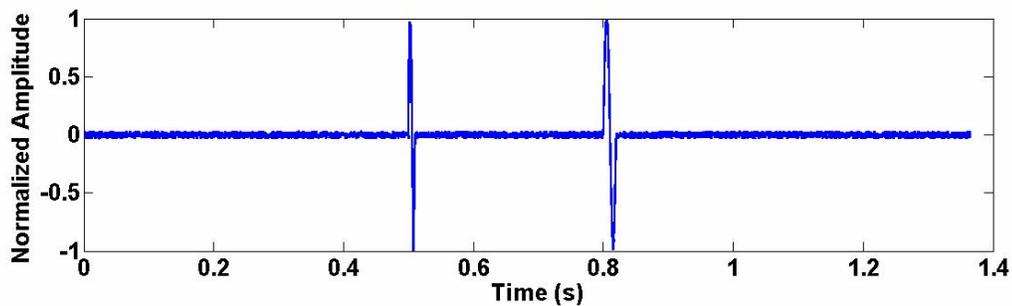


Figure 1.22: Synthetic “good” trace with SNR = 30 dB.

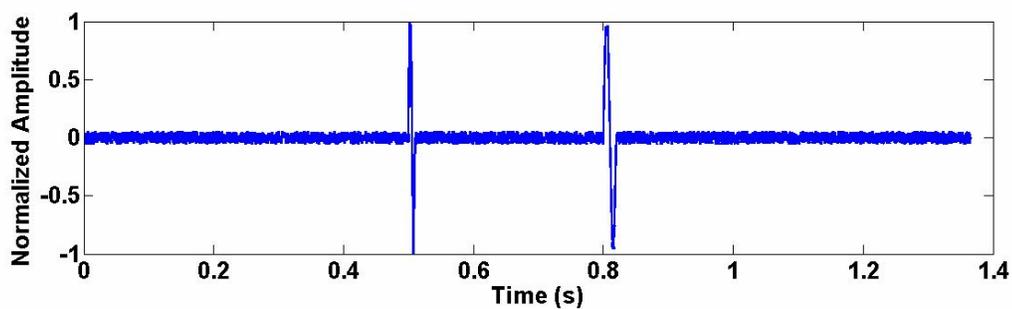


Figure 1.23: Synthetic “good” trace with SNR = 25 dB.

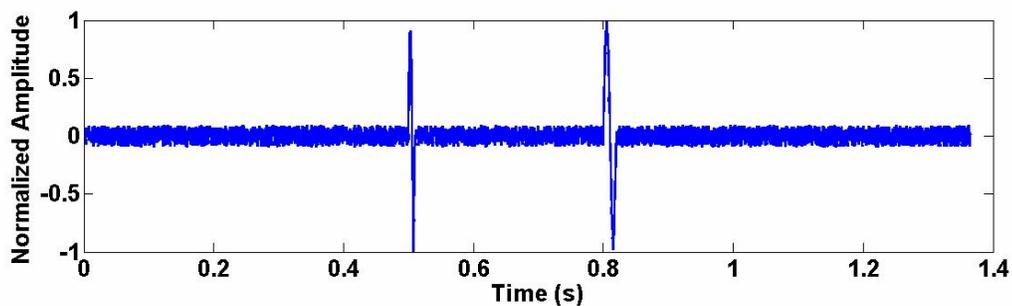


Figure 1.24: Synthetic “good” trace with SNR = 20 dB.

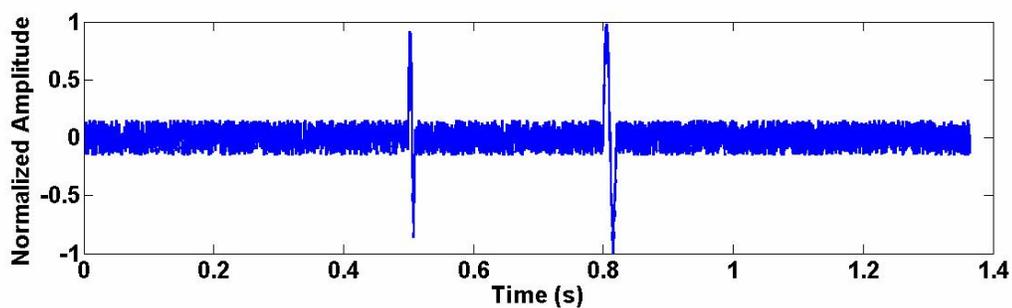


Figure 1.25: Synthetic “good” trace with SNR = 15 dB.

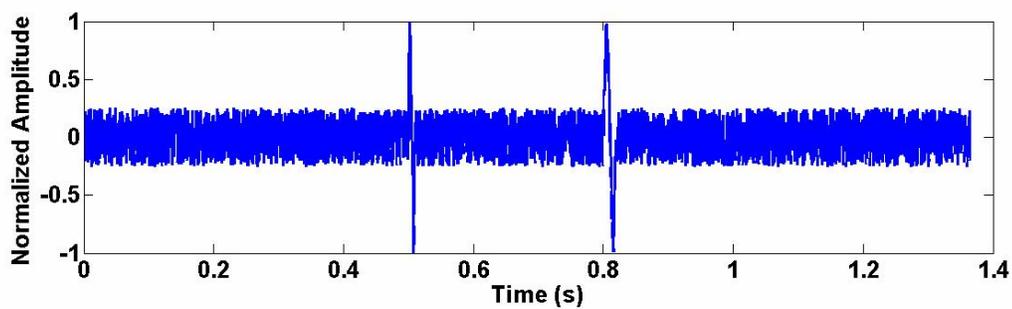


Figure 1.26: Synthetic “good” trace with SNR = 10 dB.

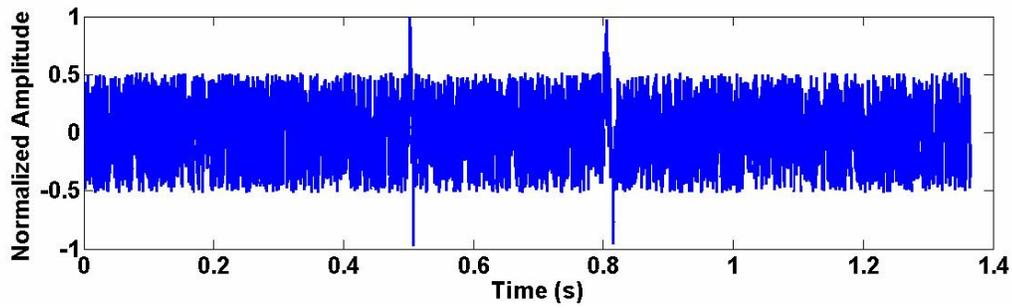


Figure 1.27: Synthetic “good” trace with SNR = 0 dB.

### 1.3.2 Noise Event

A synthetic noise event with a duration of 1 second,  $s_n(t)$ , was created. Referring to equation 1.7 with  $A_n=1$ , the synthetic the noise event is expressed as

$$\begin{aligned} s_n(t) &= n(t), & 0.2 \text{ s} \leq t \leq 1.2 \text{ s}; \\ s_n(t) &= 0, & \text{otherwise.} \end{aligned} \quad (1.11)$$

This noise event is static shifted to have zero mean and is normalized between an amplitude range of -1 to 1. This event is depicted in Figure 1.28.

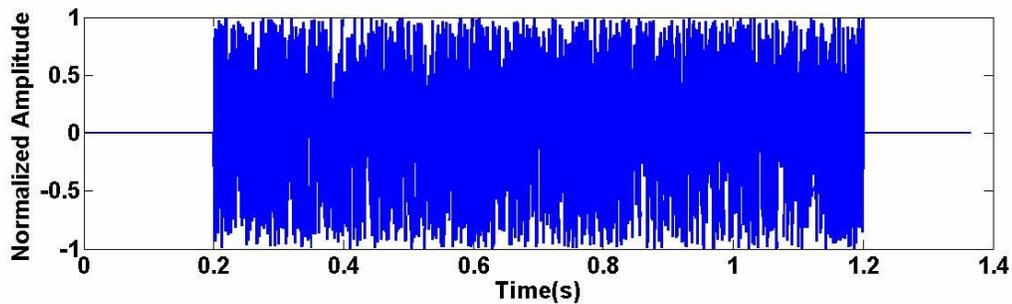


Figure 1.28: Synthetic noise trace.

## **1.4 Thesis Objective**

The current event-file classification software has been known to misclassify a large portion of the received files. This has resulted in many noise events being incorrectly identified as “good” events, often referred to as “false positives”. These numerous misclassifications require extensive manual investigation. This time-consuming process of examining incorrectly classified files one-by-one can become very costly.

The purpose was to develop and combine microseismic signal analysis algorithms capable of precisely classifying the microseismic event files generated by the passive-seismic monitoring system at Cold Lake. Compared to noise events, I observed that many "good" events generally have lower dominant frequency content, shorter P-wave event-lengths, and flatter time-domain characteristics. Based on these observations, classification algorithms involving frequency-filtering, event-length detection, and statistical analysis were developed. In the passive-seismic monitoring industry, it is often the convention to examine microseismic signal amplitudes for event-classification; however, the purpose here was to develop novel classification techniques that are more robust than simple amplitude examination.

## **1.5 Original Thesis Contributions**

Novel passive-seismic event-classification algorithms involving frequency-filtering, event-length detection, and statistical analysis were created through observation, research, development, and optimization. In this thesis, theoretical background pertaining to developed classification algorithms will be provided in addition to application examples. A MATLAB<sup>®</sup> graphical user interface was created that combined many of these algorithms

into a self-contained application. Following this, extensive testing was performed to determine which classification algorithms were most accurate. These tests revealed that the developed statistical analysis classification algorithms performed best for the dataset at Cold Lake. Theoretical mathematics of principal components analysis (PCA) was explored and an application of PCA to statistical analysis algorithm output results was performed to optimize classification performance.

A second, updated, application that employs PCA on the statistical algorithm results was developed and exhaustively tested on a wide-range of microseismic files from 72 different production pads. This application yielded classification accuracies between 90% and 99.5%. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this developed application could have significant future impact.

## 1.6 Thesis Outline

Chapter one provides background regarding passive-seismic monitoring. Imperial Oil's Cold Lake heavy-oil production operations are discussed explaining the necessity of employing their passive-seismic monitoring system. My thesis objective and original contributions are provided.

Chapter two pertains to the developed frequency-filtering algorithms. An overview of frequency-filtering applications is first provided. Strengths and weaknesses of various filter frequency responses are discussed. Developed filters are applied to the microseismic traces of interest. Detailed filter theory and background are provided in Appendices A and B.

Chapter three contains discussions on the developed event-length detection algorithms. After relevant theory is provided, event-length detection algorithms are developed and applied to the microseismic signals of interest.

Chapter four examines observed statistical characteristics of "good" and noise microseismic signals. Applicable statistical theory is discussed. Statistical analysis algorithms are then developed and applied.

Chapter five concerns performing multivariate data reduction using a technique referred to as "principal components analysis". Extensive testing on a wide range of microseismic data from Cold Lake has revealed that statistical analysis algorithms yield results with the highest potential for consistently accurate microseismic file classification. Theoretical mathematics of principal components analysis is given in Appendix C. Application of this analysis to statistical algorithm output results is demonstrated.

Chapter six discusses implementation details pertaining to two cases: one where many created algorithms are combined into a graphical user interface for file classification; and the updated application which performs principal components analysis on statistical algorithm results to classify microseismic files. The latter configuration is found to yield more accurate results when applied to a wide range of data. Three major datasets are tested on this configuration and the results are provided.

Chapter seven summarizes this thesis and contains pertinent conclusions.

## CHAPTER TWO: Frequency Filtering

### 2.1 Introduction

In signal processing applications, it is often desired to pass certain frequency components of an input signal, while eliminating others. This is the essence of frequency filtering. A system can be said to filter an input signal if it presents different frequency responses to different frequency components of the signal (e.g. Haykin and Van Veen, 2003).

I observed that many "good" events generally have lower dominant frequency content than noise events. This chapter shows the application of frequency filtering to microseismic signals to demonstrate the potential of this technique regarding event-classification. Background on frequency-filtering terminology and theory is provided. The strengths and weaknesses of the Butterworth, Chebyshev, and Inverse-Chebyshev frequency response approximations are explored prior to application.

### 2.2 Background

#### 2.2.1 Terminology

Filters can be described in terms of how they process input signals. A *low-pass* filter passes lower frequency components while attenuating high-frequency components, while a *high-pass* filter passes high frequencies while attenuating low frequencies. A *band-pass* filter passes signals that lie within a certain frequency band, while attenuating signals outside that band (e.g. Haykin and Van Veen, 2003).

The range of frequencies that are passed by a filter is referred to as its *passband*, while the range of frequencies that are attenuated by a filter is referred to as its *stopband*.

Realistic and realizable filters have a smooth transition between the passband and stopband, called the *transition band*. The *order* of the filter is the order of the characteristic differential equation that describes its impulse response in the time domain. The higher the filter's order, the sharper its transition band (e.g. Maundy, 2005).

When performing frequency analysis, it is usually desired to analyze these characteristics in the frequency domain. A filter's impulse response in the time domain corresponds to its transfer function in the frequency domain. In the frequency domain, the order of a filter is the highest power of the Laplace operator " $s$ " that can be seen in the denominator of its transfer function. In real applications, as the desired accuracy of a filter increases, its required order increases as well. As the order of a filter increases, its realization increases in complexity. More components are required to construct high order filters. Thus, there is a tradeoff between filter accuracy and complexity.

### 2.2.2 Common Filter Frequency Response Approximations

In practical applications, there are four main filter responses that rely on approximations (e.g. Maundy, 2005): the Butterworth, Chebyshev, Inverse-Chebyshev, and Cauer. Three of them -- the Butterworth, Chebyshev, and Inverse-Chebyshev responses -- are used to develop passive-seismic event-classification algorithms.

Figure 2.1 (adapted from Maundy, 2005) contains sketches of the attenuation characteristics of these four filter types for low-pass filtering. In Figure 2.1,  $\omega_p$  and  $\omega_s$  represent the passband and stopband limits, respectively, in radians per second. The quantity  $\alpha(\omega)$  signifies the attenuation of the filter (in dB) as a function of the angular

frequency. The symbols  $\alpha_{\min}$  and  $\alpha_{\max}$  are the desired minimum and maximum allowable attenuations in the stopband and passband, respectively.

For high-pass filtering, the attenuation curve in Figure 2.1 would start high and decrease with increasing frequency. Also, the positions of  $\alpha_{\min}$  and  $\alpha_{\max}$  on the  $\omega$  axis would be interchanged on Figure 2.1, as well as the positions of  $\omega_s$  and  $\omega_p$  (high pass filters attenuate low frequencies).

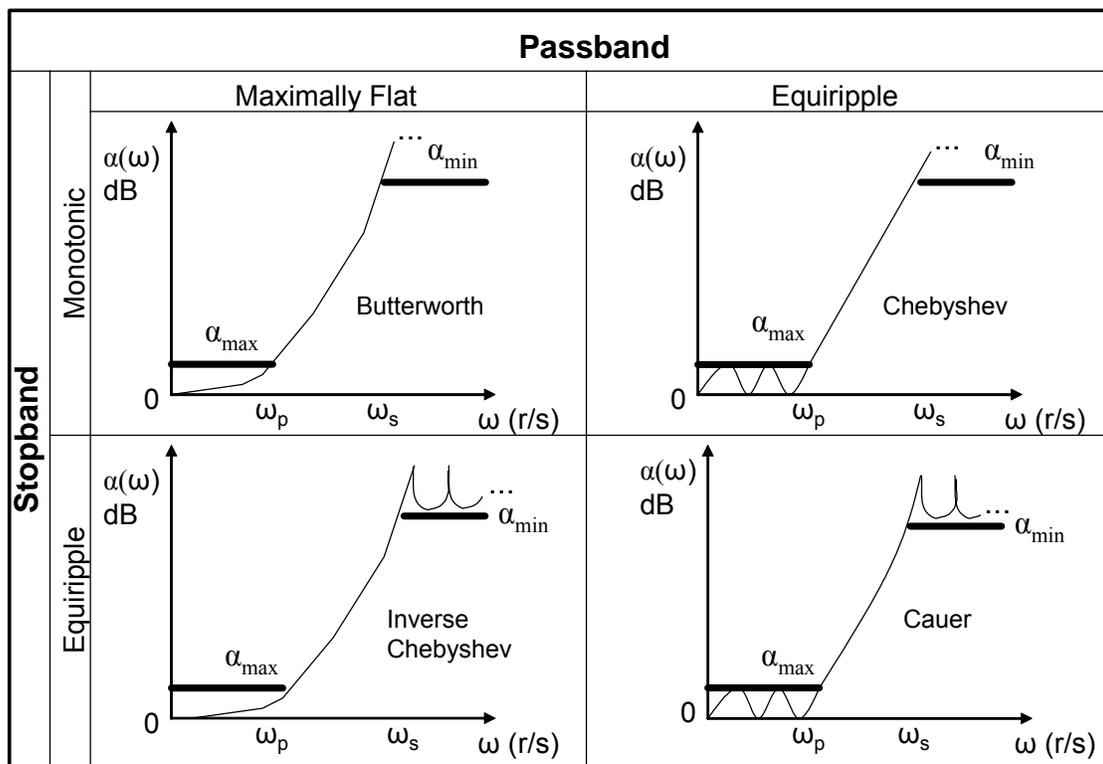


Figure 2.1: Sketches of attenuation characteristics of 4 main filter responses for a low-pass filter case: Butterworth (top-left), Chebyshev (top-right), Inverse Chebyshev (bottom-left), and Cauer (bottom-right). The horizontal axis  $\omega$  pertains to angular frequency measured in radians per second. The vertical axis,  $\alpha(\omega)$ , represents the filter's loss function measured in decibels (dB) (adapted from Maundy, 2005).

As shown in Figure 2.1, the Butterworth response has a maximally flat attenuation characteristic in the passband as frequency increases (an advantage) and a monotonically increasing attenuation characteristic in the stopband (e.g. Maundy, 2005). The transition

region between the passband and stopband limits of the Butterworth filter, however, is less sharp, a disadvantage, than that of a different response, such as the Chebyshev.

The Chebyshev response has an *equiripple* attenuation characteristic in the passband (e.g. Zhou and McMechan, 1999) and a monotonically increasing attenuation characteristic in the stopband. The passband equiripple characteristic pertains to oscillations in the frequency response, which is a disadvantage of the Chebyshev filter; however, this response provides greater stopband attenuation than the Butterworth response for a given filter order, which is advantageous (e.g. Maundy, 2005).

The Inverse-Chebyshev response contains a maximally flat attenuation characteristic in the passband and an equiripple characteristic in the stopband. For a given minimum stopband attenuation, an Inverse-Chebyshev filter will require a lower order than a Butterworth filter (an advantage). When designing band-elimination topologies, however, the Inverse-Chebyshev response requires circuit realizations with high complexity (e.g. Maundy, 2005).

To provide a diverse set of frequency responses and optimize the aggregate performance of these classification algorithms, each of the Butterworth, Chebyshev, and Inverse-Chebyshev frequency response approximations were modeled as filters in MATLAB<sup>®</sup>.

Practical filters require a smooth transition between the passband and stopband to be realizable. A smooth transition band is also required to minimize the effects from Gibbs' Phenomenon, which states that if a sharp cutoff is applied in the frequency domain, undesired oscillations, or "ringing", will occur in the time domain (e.g. Sheriff and Geldart, 1995). To minimize this ringing, a smooth transition band is required.

A preliminary available dataset of 7032 microseismic event files was used to decide algorithm details such as filter order, passband range, and stopband range.

### 2.3 Low-Pass Inverse-Chebyshev Filter

A fourth-order ( $n=4$ ) Inverse-Chebyshev low-pass filter with a stopband edge angular frequency ( $\omega_s$ ) of 628 rad/s (corresponding to a frequency of 100 Hz), passband edge frequency ( $\omega_p$ ) of 225 rad/s (35.8 Hz), maximum passband attenuation ( $\alpha_{max}$ ) of 0.25 dB, and minimum stopband attenuation ( $\alpha_{min}$ ) of 40 dB was implemented as a classification technique. Detailed development of the Inverse-Chebyshev low-pass filter frequency response is shown in Appendix A. Figure 2.2 shows the magnitude response for this filter generated in MATLAB<sup>®</sup>.

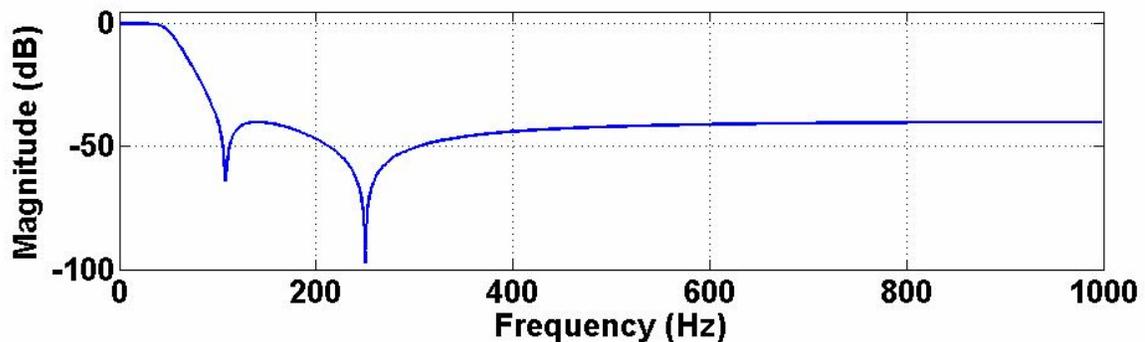


Figure 2.2: Amplitude response of low-pass Inverse-Chebyshev filter.

Figures 2.3 to 2.8 show the synthetic “good” traces depicted in Figures 1.22 to 1.27 after applying this Inverse-Chebyshev low-pass filter. Figure 2.9 shows the synthetic noise trace depicted in Figure 1.28 after filtering. Compared to the peak amplitude of the filtered noise synthetic, the largest peak amplitude discrepancy when examining filtered “good” synthetics pertains to the case where the “good” synthetic with the highest SNR was input

to the filter. This can be seen by comparing Figure 2.3, which corresponds to an input synthetic “good” trace with SNR = 30 dB, to Figure 2.9, which corresponds to the synthetic noise input. The peak amplitude of the signal shown in Figure 2.3 is approximately 0.4, over twice the approximate peak amplitude of the signal shown in Figure 2.9, which is approximately 0.15. As the SNR of the input synthetic “good” trace decreases, the peak amplitude discrepancy between the filtered “good” and noise synthetic traces decreases as well. This discrepancy decrease can be seen by comparing Figures 2.4 through 2.8 to Figure 2.9. Table 2.1 summarizes the peak amplitudes of the synthetic traces after applying this low-pass Inverse-Chebyshev filter.

Figures 2.10 to 2.19 show the “good” events from Figures 1.7 to 1.16, respectively, after low-pass filtering. Figures 2.20 to 2.24 show the noise events from Figures 1.17 to 1.21, respectively, after filtering. Table 2.2 summarizes the filtered peak amplitudes of these traces. Note that the magnitudes of the filtered “good” events are considerably higher than those of the filtered noise events.

Table 2.1: Summary of peak amplitudes of synthetic traces after applying low-pass Inverse-Chebyshev filter.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Peak Amplitude</b>
Good	30	0.37
Good	25	0.35
Good	20	0.31
Good	15	0.24
Good	10	0.21
Good	0	0.21
Noise	N/A	0.16

Table 2.2: Summary of peak amplitudes of “good” and noise events after applying low-pass Inverse-Chebyshev filter. Note that the magnitudes of the filtered “good” events are considerably higher than those of the filtered noise events.

Event Type	Event #	Peak Amplitude
Good	1	0.47
Good	2	0.35
Good	3	0.33
Good	4	0.28
Good	5	0.33
Good	6	0.57
Good	7	0.55
Good	8	0.22
Good	9	0.54
Good	10	0.26
Noise	11	0.13
Noise	12	0.037
Noise	13	0.17
Noise	14	0.18
Noise	15	0.084

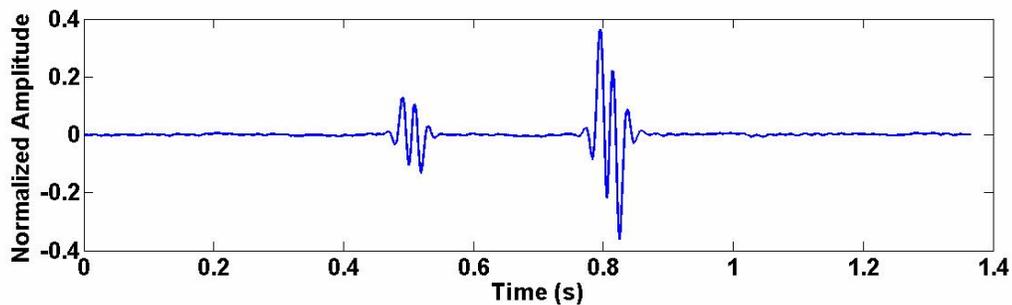


Figure 2.3: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 30 dB.

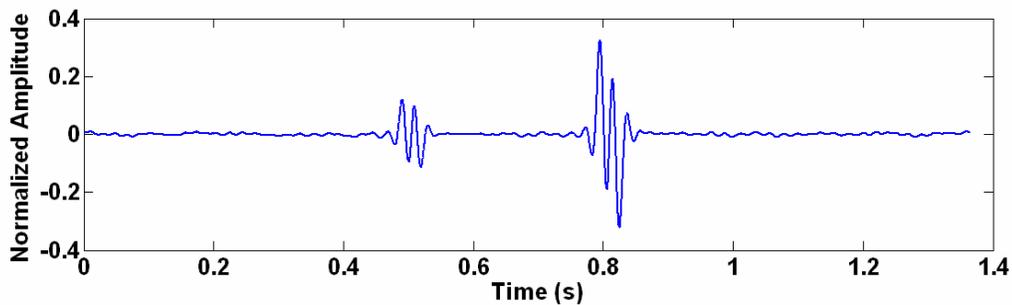


Figure 2.4: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 25 dB.

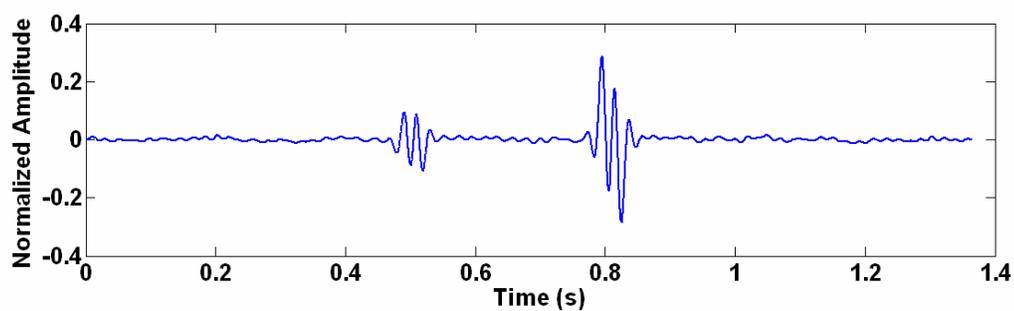


Figure 2.5: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 20 dB.

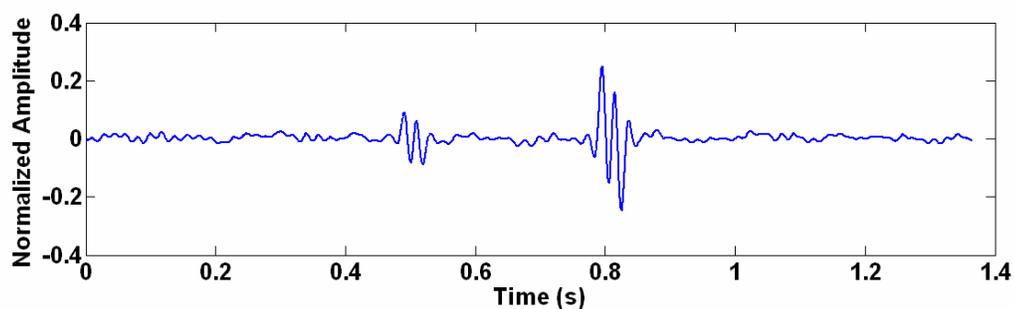


Figure 2.6: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 15 dB.

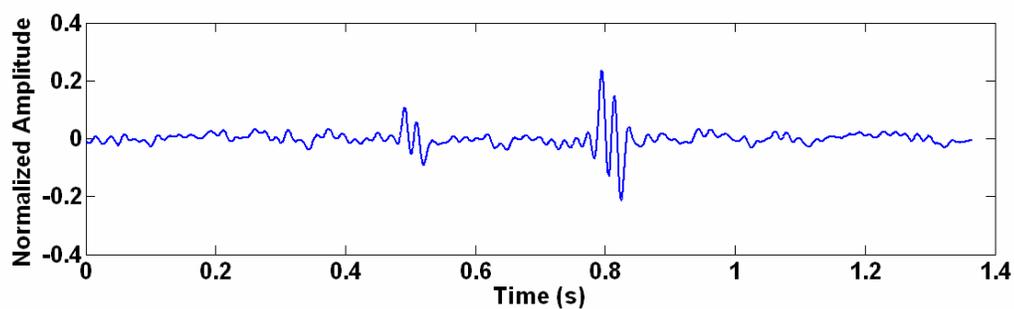


Figure 2.7: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 10 dB.

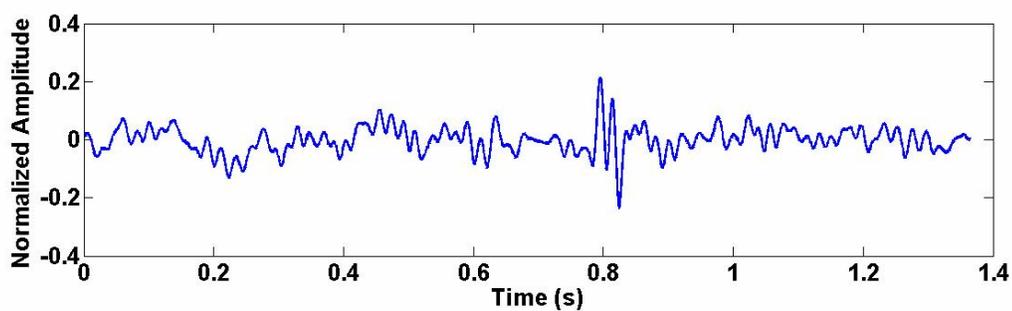


Figure 2.8: Low-pass Inverse-Chebyshev filter applied to synthetic “good” trace shown in Figure 1.27 with SNR = 0 dB.

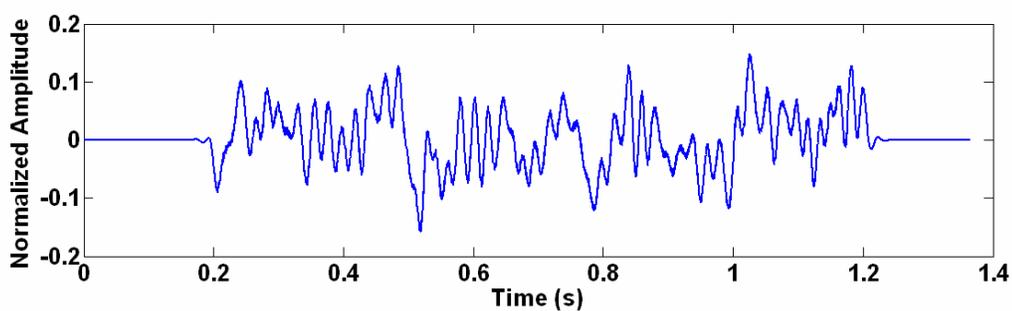


Figure 2.9: Low-pass Inverse-Chebyshev filter applied to synthetic noise trace shown in Figure 1.28.

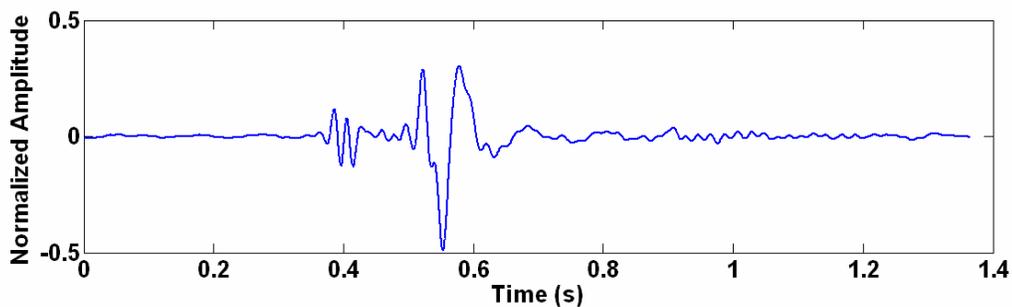


Figure 2.10: Low-pass Inverse-Chebyshev filter applied to Event #1, a “good” event.

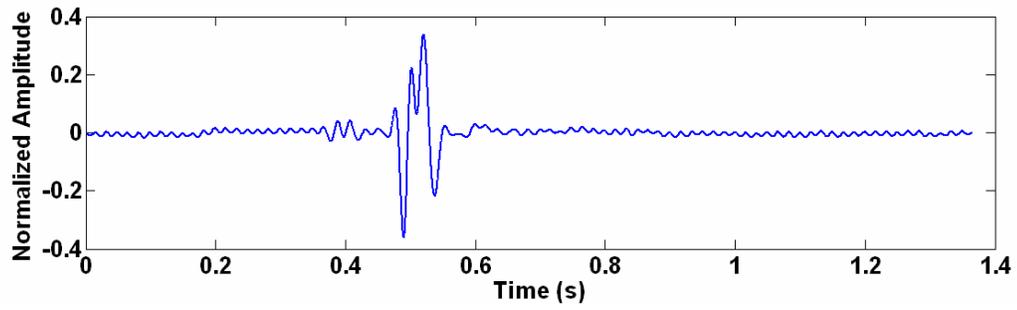


Figure 2.11: Low-pass Inverse-Chebyshev filter applied to Event #2, a “good” event.

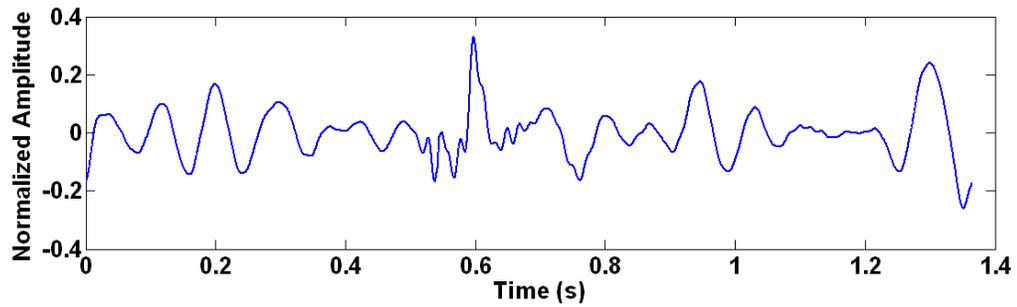


Figure 2.12: Low-pass Inverse-Chebyshev filter applied to Event #3, a “good” event.

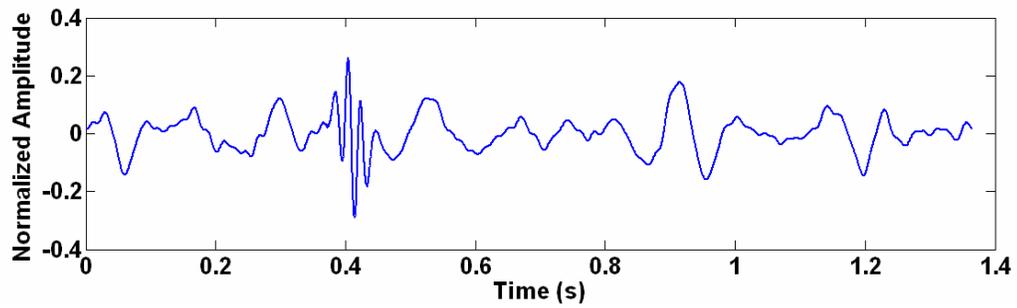


Figure 2.13: Low-pass Inverse-Chebyshev filter applied to Event #4, a “good” event.

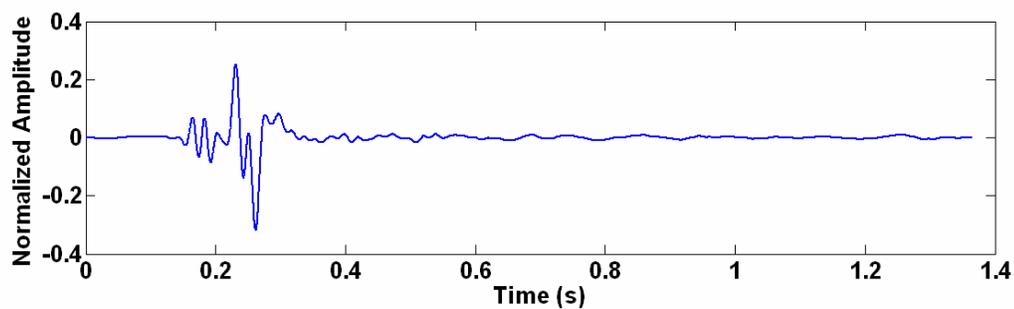


Figure 2.14: Low-pass Inverse-Chebyshev filter applied to Event #5, a “good” event.

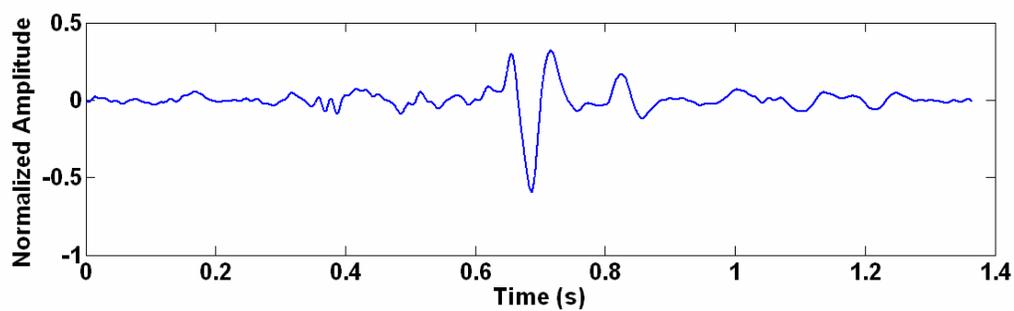


Figure 2.15: Low-pass Inverse-Chebyshev filter applied to Event #6, a “good” event.

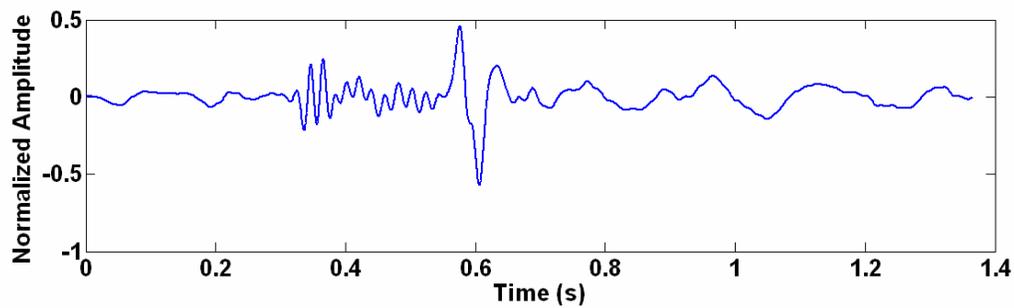


Figure 2.16: Low-pass Inverse-Chebyshev filter applied to Event #7, a “good” event.

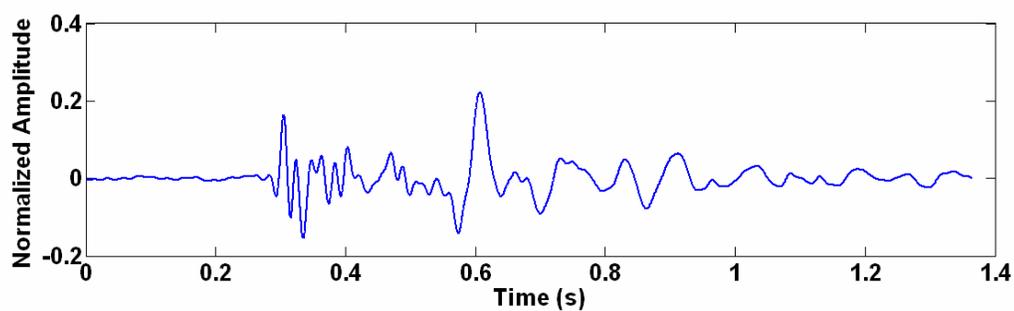


Figure 2.17: Low-pass Inverse-Chebyshev filter applied to Event #8, a “good” event.

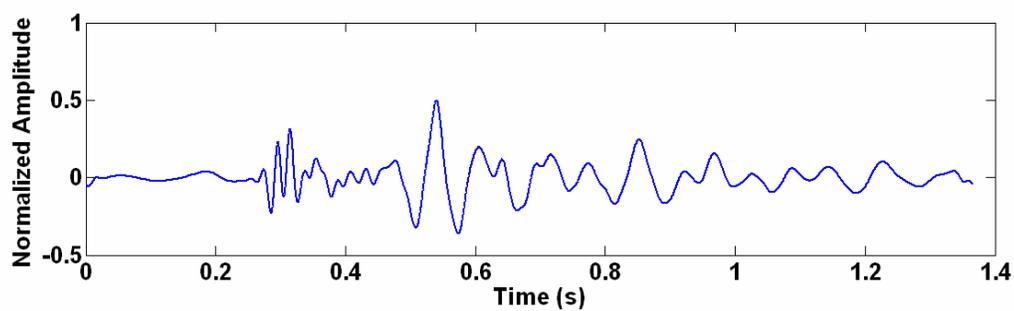


Figure 2.18: Low-pass Inverse-Chebyshev filter applied to Event #9, a “good” event.

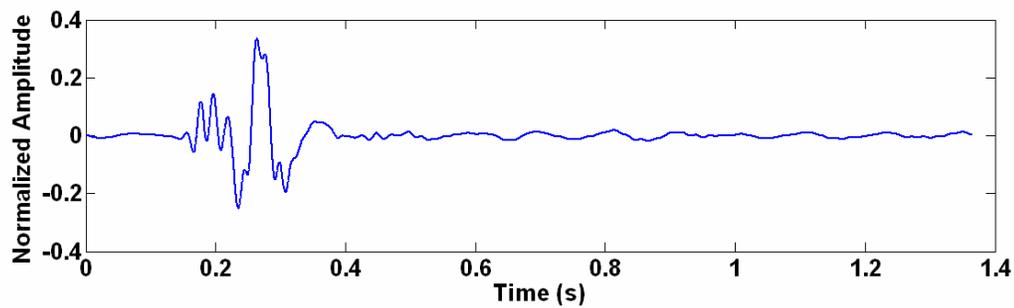


Figure 2.19: Low-pass Inverse-Chebyshev filter applied to Event #10, a “good” event.

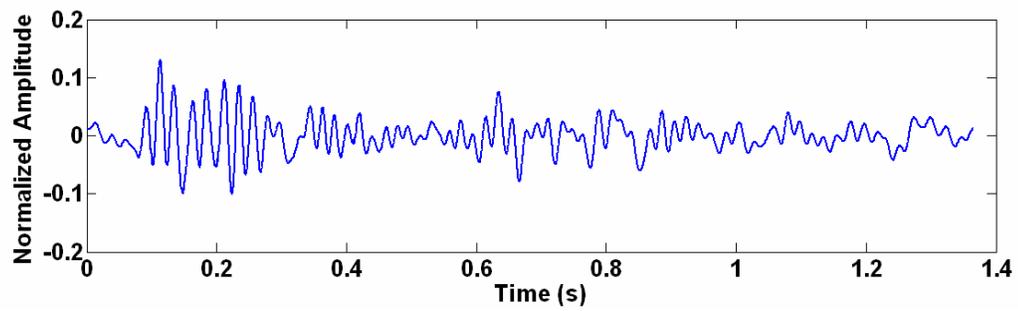


Figure 2.20: Low-pass Inverse-Chebyshev filter applied to Event #11, a noise event.

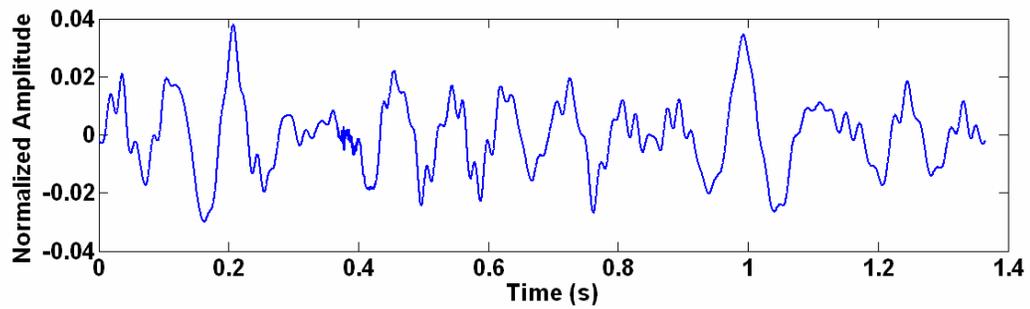


Figure 2.21: Low-pass Inverse-Chebyshev filter applied to Event #12, a noise event.

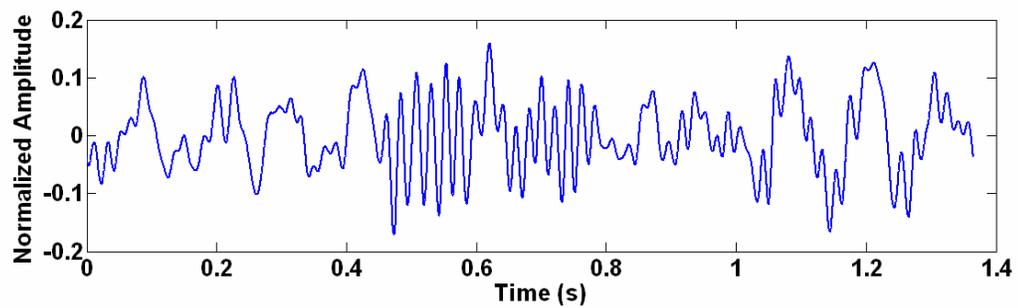


Figure 2.22: Low-pass Inverse-Chebyshev filter applied to Event #13, a noise event.

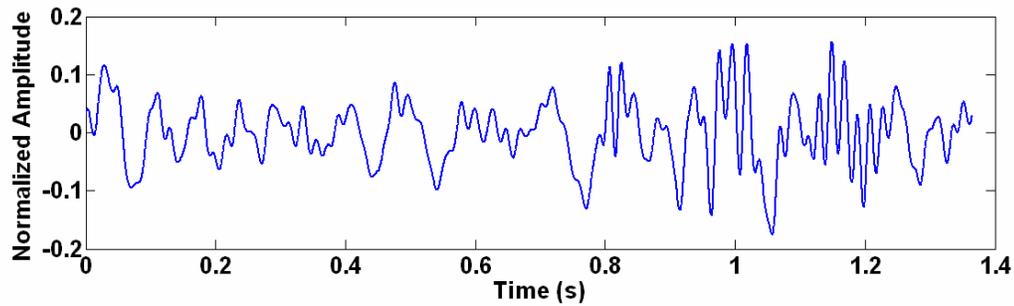


Figure 2.23: Low-pass Inverse-Chebyshev filter applied to Event #14, a noise event.

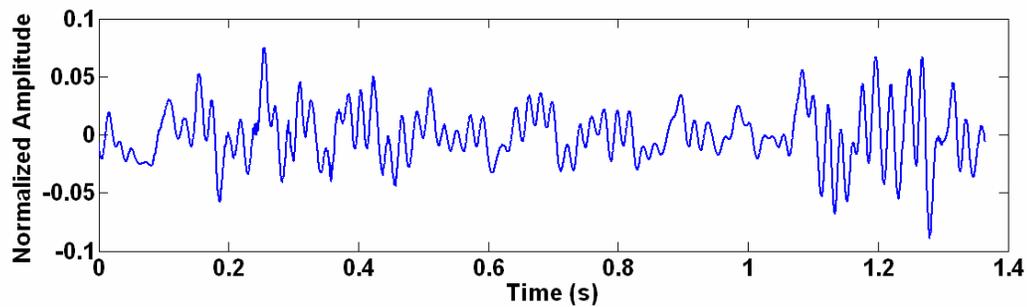


Figure 2.24: Low-pass Inverse-Chebyshev filter applied to Event #15, a noise event.

## 2.4 High-Pass Butterworth Filter

A fourth-order ( $n=4$ ) Butterworth high-pass filter with a passband edge angular frequency ( $\omega_p$ ) of 2500 rad/s (398 Hz), stopband edge frequency ( $\omega_s$ ) of 1351 rad/s (215 Hz), maximum passband attenuation ( $\alpha_{max}$ ) of 3 dB, and minimum stopband attenuation ( $\alpha_{min}$ ) of 25 dB was implemented as a classification technique. Detailed development of the Butterworth high-pass filter frequency response is shown in Appendix B. Figure 2.25 shows the magnitude response for this filter generated in MATLAB<sup>®</sup>.

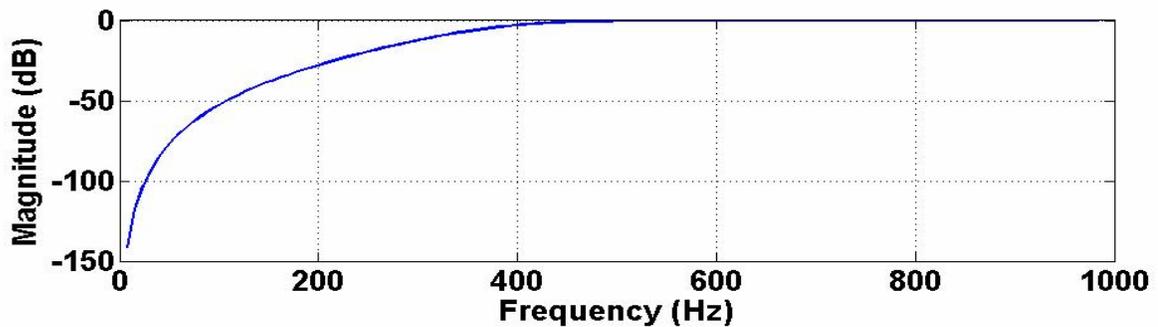


Figure 2.25: Amplitude response of high-pass Butterworth filter.

Figures 2.26 to 2.31 show the synthetic “good” traces depicted in Figures 1.22 to 1.27 after applying this Butterworth high-pass filter. Figure 2.32 shows the synthetic noise trace depicted in Figure 1.28 after filtering. Compared to the peak amplitude of the filtered noise synthetic, the largest peak amplitude discrepancy when examining filtered “good” synthetics pertains to the case where the “good” synthetic with the highest SNR was input to the filter. This can be seen by comparing Figure 2.26, which corresponds to an input synthetic “good” trace with SNR = 30 dB, to Figure 2.32, which corresponds to the synthetic noise input. The peak amplitude of the signal shown in Figure 2.26 is less than 0.05, significantly less than the peak amplitude of the signal shown in Figure 2.32, which is approximately 0.9. As the SNR of the input synthetic “good” trace decreases, the peak amplitude discrepancy between the filtered “good” and noise synthetic traces decreases as well. This discrepancy decrease can be seen by comparing Figures 2.27 through 2.31 to Figure 2.32. Table 2.3 summarizes the peak amplitudes of the synthetic traces after applying this high-pass Butterworth filter.

Figures 2.33 to 2.42 show the “good” events from Figures 1.7 to 1.16, respectively, after high-pass filtering. Figures 2.43 to 2.47 show the noise events from Figures 1.17 to 1.21, respectively, after filtering. Table 2.4 summarizes the filtered peak amplitudes of

these traces. Note that the magnitudes of the filtered “good” events are considerably lower than those of the filtered noise events.

Table 2.3: Summary of peak amplitudes of synthetic traces after applying high-pass Butterworth filter.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Peak Amplitude</b>
Good	30	0.045
Good	25	0.055
Good	20	0.1
Good	15	0.14
Good	10	0.23
Good	0	0.49
Noise	N/A	0.91

Table 2.4: Summary of peak amplitudes of “good” and noise events after applying high-pass Butterworth filter. Note that the magnitudes of the filtered “good” events are considerably lower than those of the filtered noise events.

<b>Event Type</b>	<b>Event #</b>	<b>Peak Amplitude</b>
Good	1	0.047
Good	2	0.083
Good	3	0.262
Good	4	0.255
Good	5	0.05
Good	6	0.018
Good	7	0.013
Good	8	0.0063
Good	9	0.028
Good	10	0.016
Noise	11	0.74
Noise	12	0.40
Noise	13	0.27
Noise	14	0.31
Noise	15	0.59

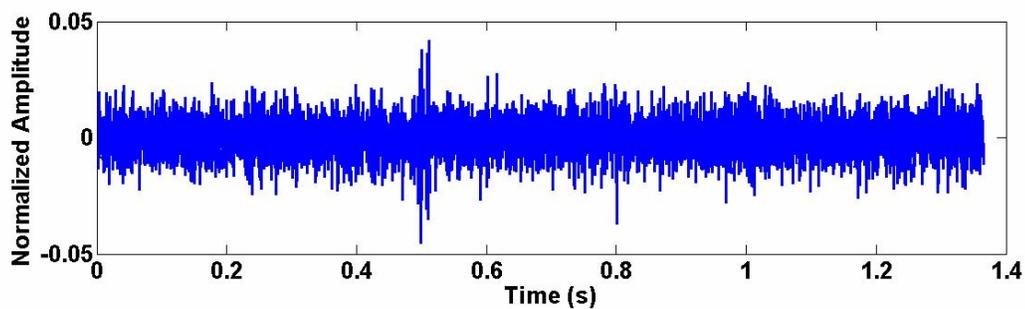


Figure 2.26: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 30 dB.

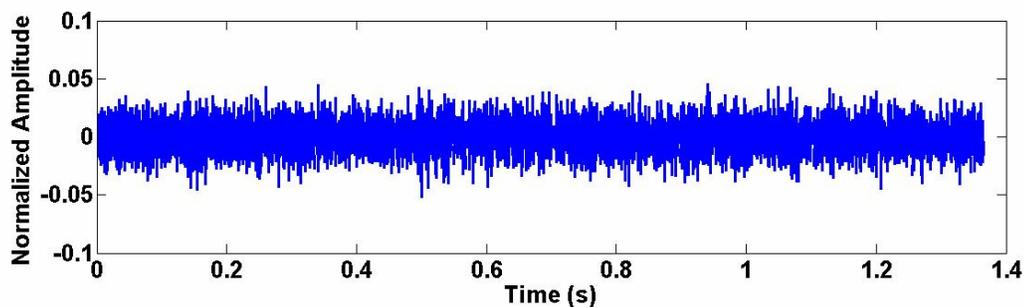


Figure 2.27: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 25 dB.

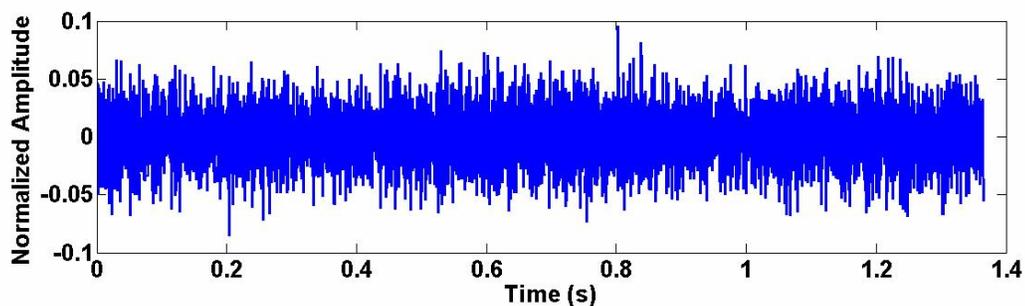


Figure 2.28: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 20 dB.

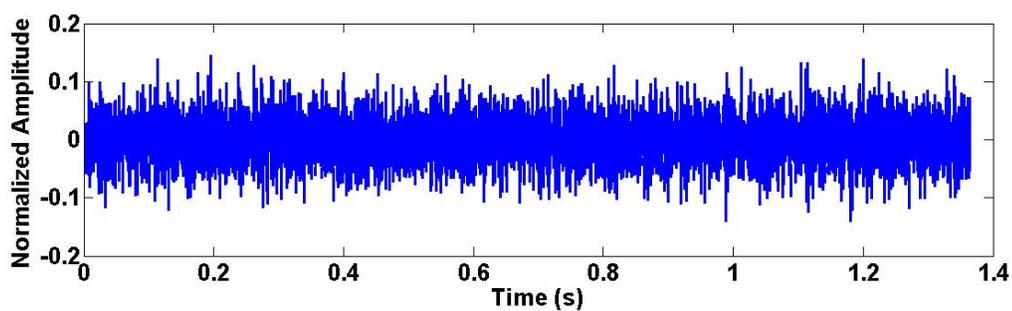


Figure 2.29: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 15 dB.

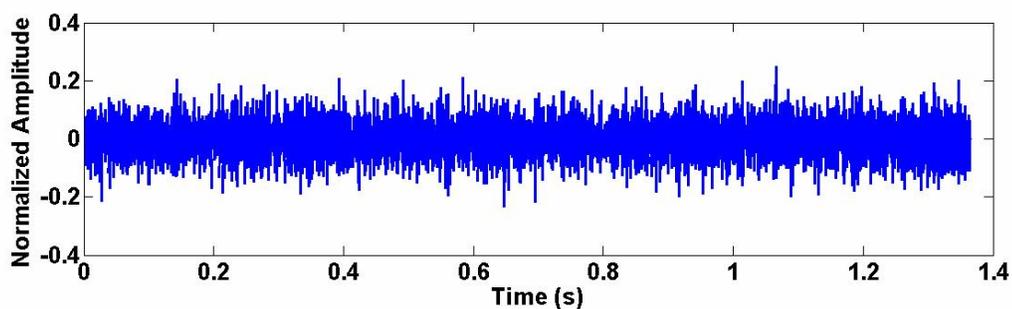


Figure 2.30: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 10 dB.

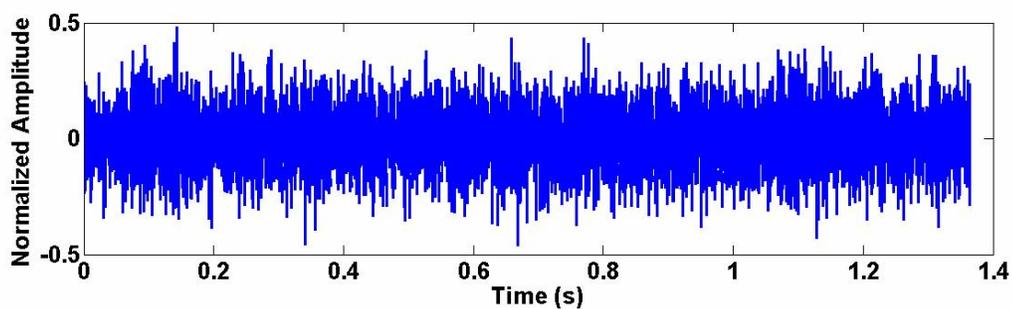


Figure 2.31: High-pass Butterworth filter applied to synthetic “good” trace shown in Figure 1.27 with SNR = 0 dB.

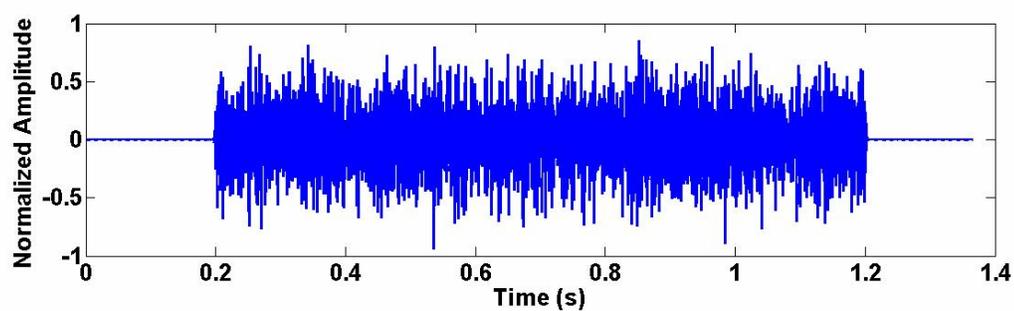


Figure 2.32: High-pass Butterworth filter applied to synthetic noise trace shown in Figure 1.28.

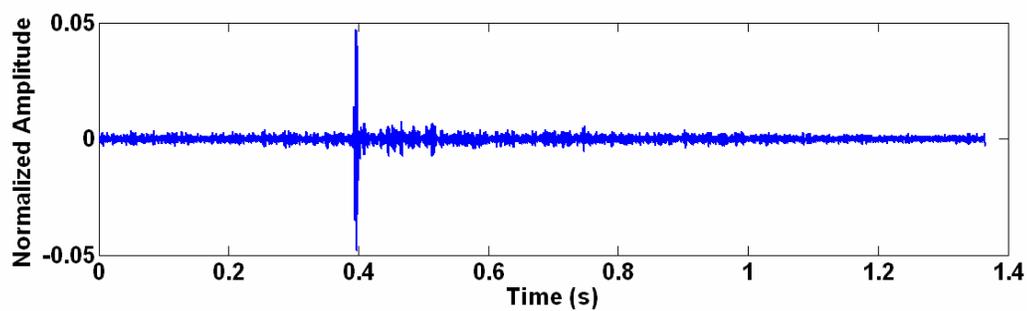


Figure 2.33: High-pass Butterworth filter applied to Event #1, a “good” event.

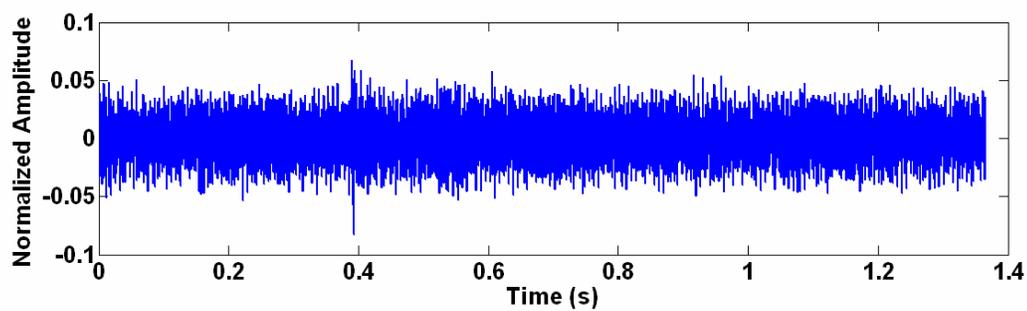


Figure 2.34: High-pass Butterworth filter applied to Event #2, a “good” event.

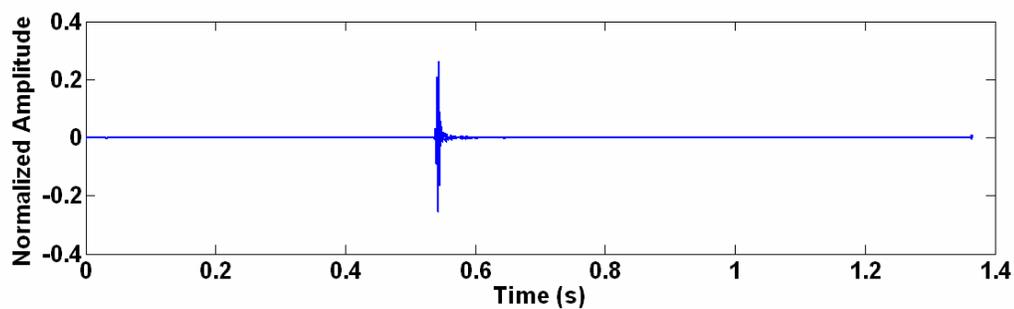


Figure 2.35: High-pass Butterworth filter applied to Event #3, a “good” event.

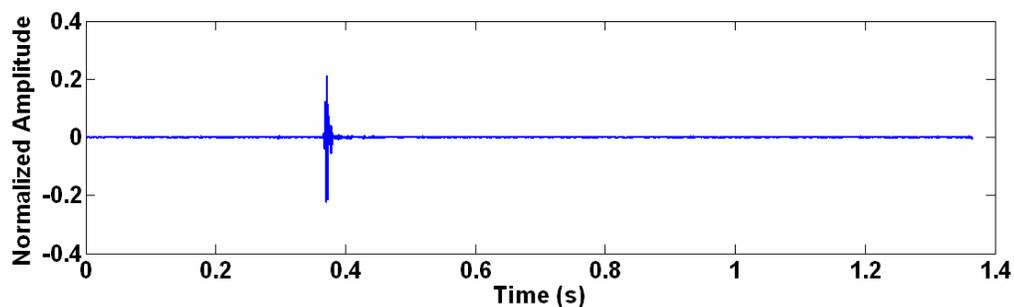


Figure 2.36: High-pass Butterworth filter applied to Event #4, a “good” event.

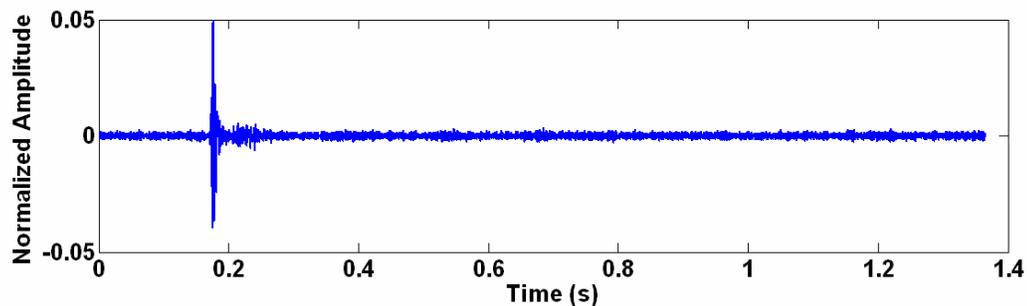


Figure 2.37: High-pass Butterworth filter applied to Event #5, a “good” event.

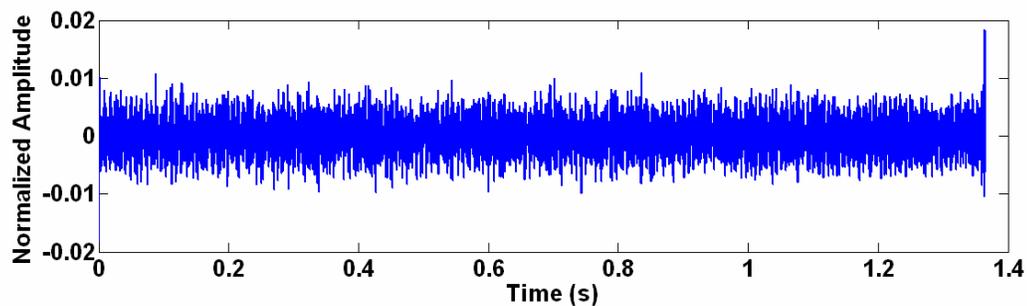


Figure 2.38: High-pass Butterworth filter applied to Event #6, a “good” event.

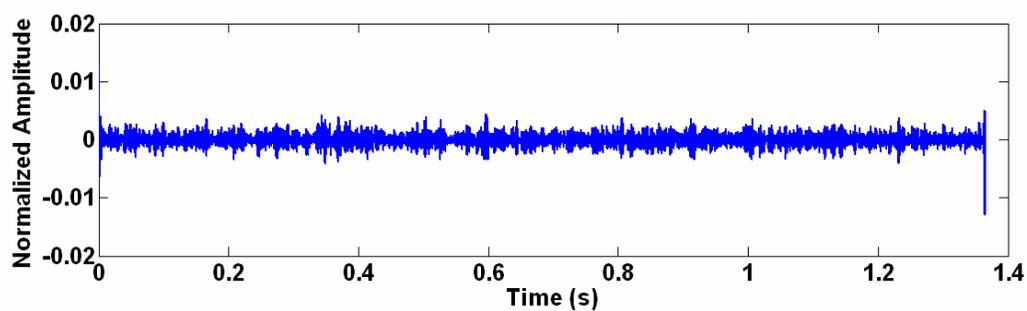


Figure 2.39: High-pass Butterworth filter applied to Event #7, a “good” event.

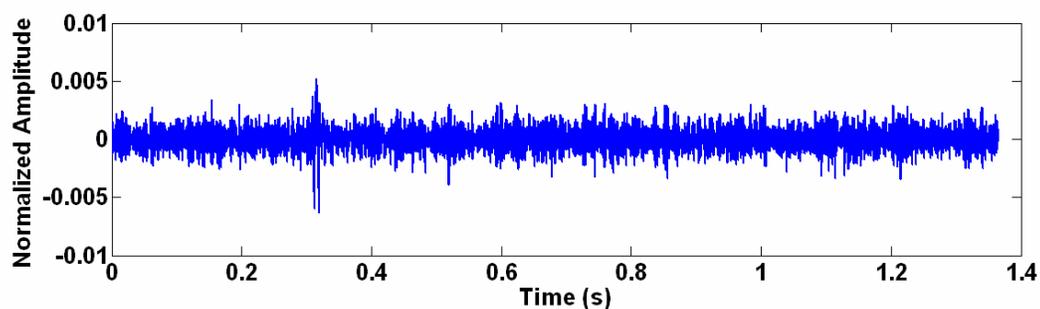


Figure 2.40: High-pass Butterworth filter applied to Event #8, a “good” event.

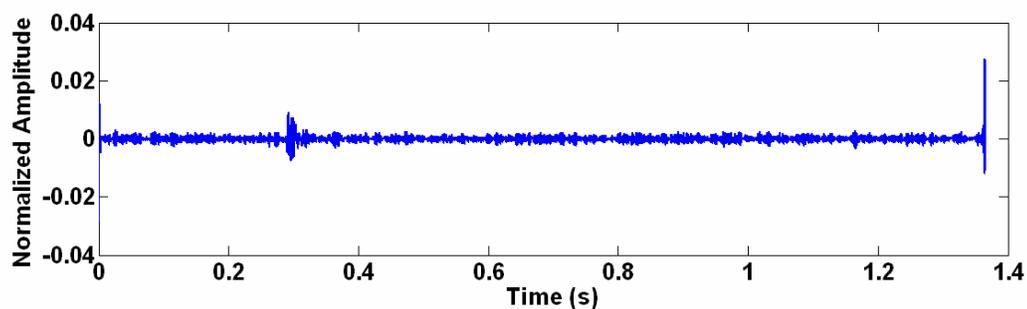


Figure 2.41: High-pass Butterworth filter applied to Event #9, a “good” event.

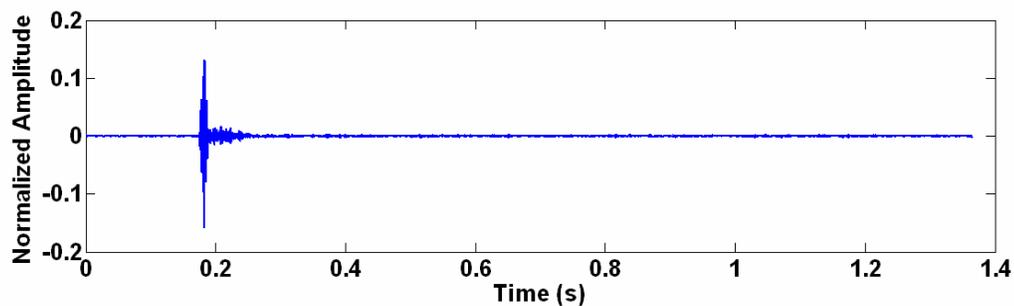


Figure 2.42: High-pass Butterworth filter applied to Event #10, a “good” event.

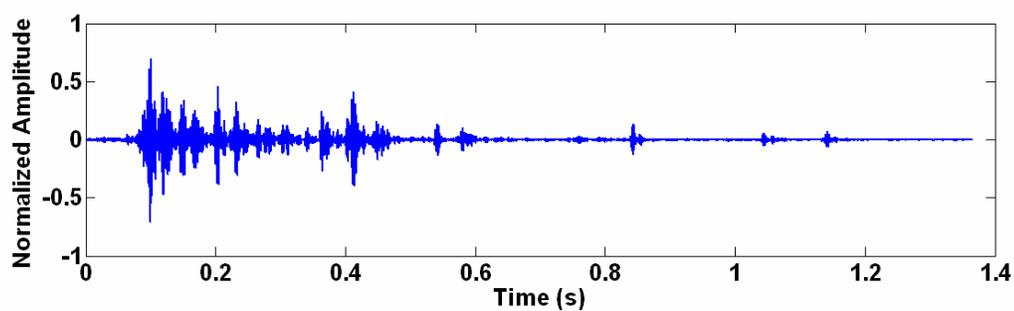


Figure 2.43: High-pass Butterworth filter applied to Event #11, a noise event.

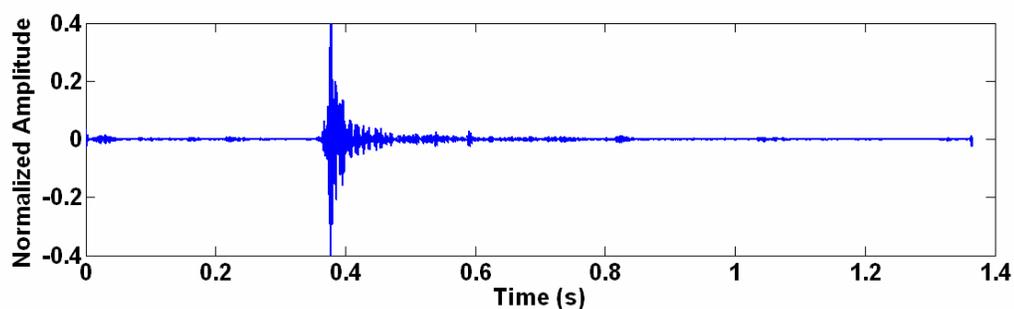


Figure 2.44: High-pass Butterworth filter applied to Event #12, a noise event.

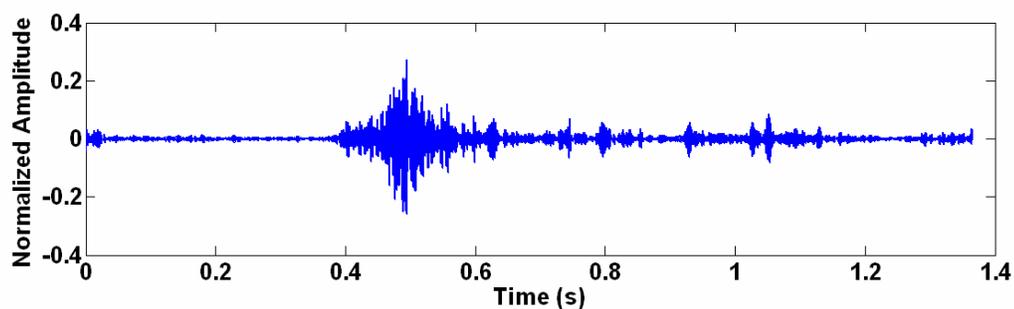


Figure 2.45: High-pass Butterworth filter applied to Event #13, a noise event.

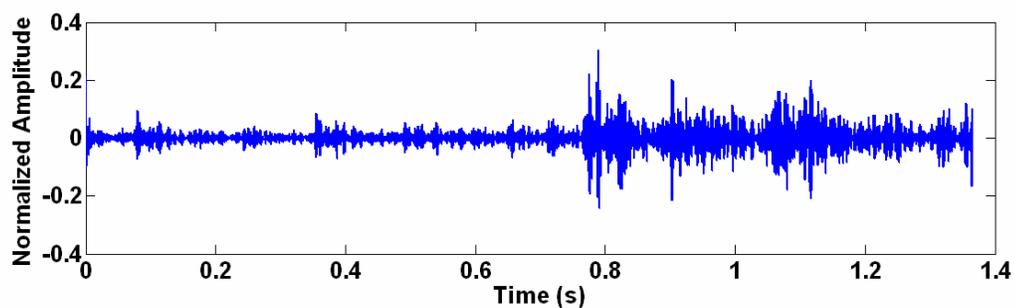


Figure 2.46: High-pass Butterworth filter applied to Event #14, a noise event.

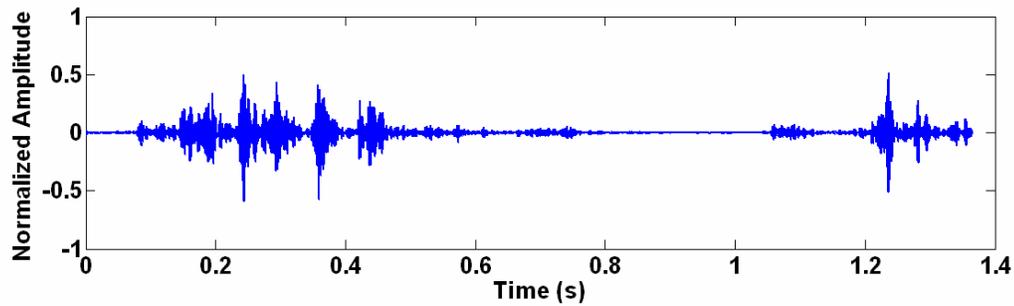


Figure 2.47: High-pass Butterworth filter applied to Event #15, a noise event.

## 2.5 Band-Pass Chebyshev Filter

A fourth-order ( $n=4$ ) Chebyshev band-pass filter with a high-frequency passband from 1000 rad/s to 2000 rad/s (159 Hz to 318 Hz) with a maximum passband attenuation ( $\alpha_{max}$ ) of 2 dB was implemented as a classification technique. The Chebyshev approximation theory is very similar to that of the Inverse-Chebyshev approximation; consequently, it is not shown here. Figure 2.48 shows the amplitude response of this filter.

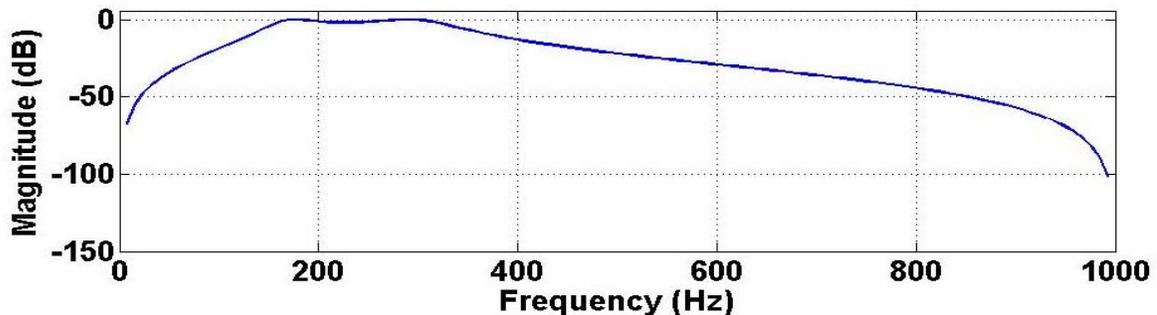


Figure 2.48: Amplitude response of band-pass Chebyshev filter.

Figures 2.49 to 2.54 show the synthetic “good” traces depicted in Figures 1.22 to 1.27 after applying this Chebyshev band-pass filter. Figure 2.55 shows the synthetic noise trace depicted in Figure 1.28 after filtering. Compared to the peak amplitude of the filtered noise synthetic, the largest peak amplitude discrepancy when examining filtered “good”

synthetics pertains to the case where the “good” synthetic with the highest SNR was input to the filter. This can be seen by comparing Figure 2.49, which corresponds to an input synthetic “good” trace with SNR = 30 dB, to Figure 2.55, which corresponds to the synthetic noise input. The peak amplitude of the signal shown in Figure 2.49 is approximately 0.12, significantly less than the peak amplitude of the signal shown in Figure 2.55, which is approximately 0.4. As the SNR of the input synthetic “good” trace decreases, the peak amplitude discrepancy between the filtered “good” and noise synthetic traces decreases as well. This discrepancy can be seen by comparing Figures 2.50 through 2.54 to Figure 2.55. Table 2.5 summarizes the peak amplitudes of the synthetic traces after applying this band-pass Chebyshev filter.

Figures 2.56 to 2.65 show the “good” events from Figures 1.7 to 1.16, respectively, after high-pass filtering. Figures 2.66 to 2.70 show the noise events from Figures 1.17 to 1.21, respectively, after filtering. Table 2.6 summarizes the filtered peak amplitudes of these traces. Note that the magnitudes of the filtered “good” events are generally considerably lower than those of the filtered noise events.

Table 2.5: Summary of peak amplitudes of synthetic traces after applying band-pass Chebyshev filter.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Peak Amplitude</b>
Good	30	0.12
Good	25	0.14
Good	20	0.19
Good	15	0.24
Good	10	0.33
Good	0	0.39
Noise	N/A	0.43

Table 2.6: Summary of peak amplitudes of “good” and noise events after applying band-pass Chebyshev filter. Note that the magnitudes of the filtered “good” events are generally considerably lower than those of the filtered noise events.

Event Type	Event #	Peak Amplitude
Good	1	0.17
Good	2	0.13
Good	3	0.36
Good	4	0.15
Good	5	0.29
Good	6	0.014
Good	7	0.065
Good	8	0.13
Good	9	0.14
Good	10	0.31
Noise	11	0.36
Noise	12	0.4
Noise	13	0.53
Noise	14	0.35
Noise	15	0.42

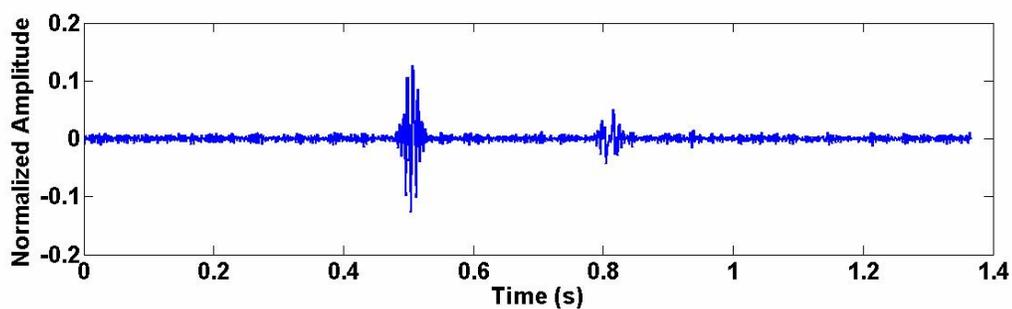


Figure 2.49: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.22 with SNR = 30 dB.

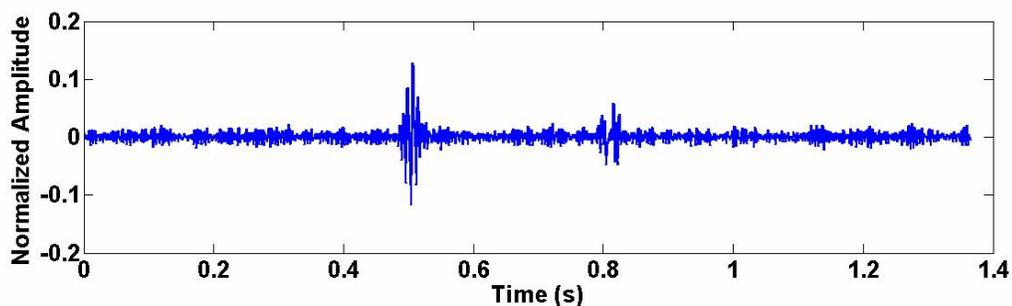


Figure 2.50: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.23 with SNR = 25 dB.

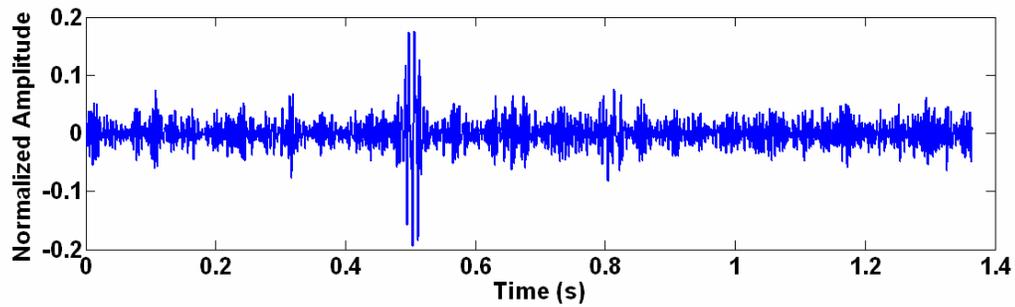


Figure 2.51: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.24 with SNR = 20 dB.

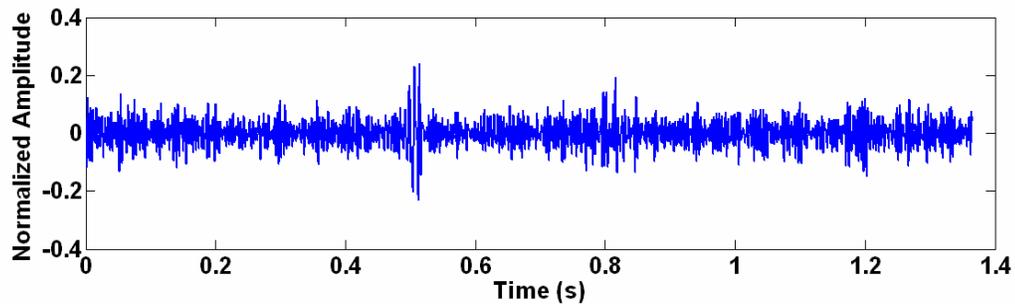


Figure 2.52: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.25 with SNR = 15 dB.

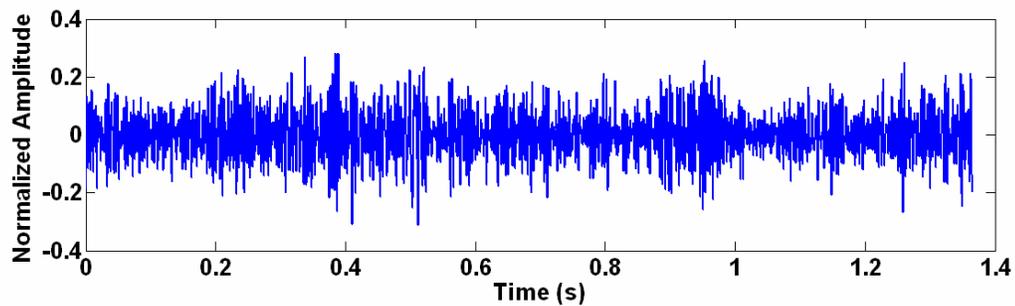


Figure 2.53: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.26 with SNR = 10 dB.

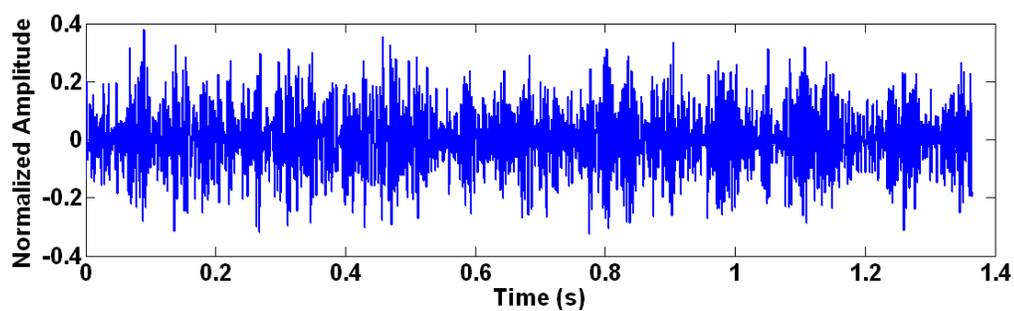


Figure 2.54: Band-pass Chebyshev filter applied to synthetic “good” trace shown in Figure 1.27 with SNR = 0 dB.

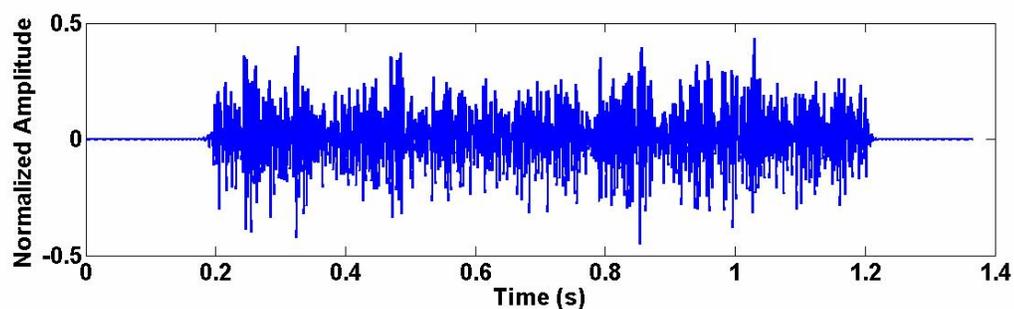


Figure 2.55: Band-pass Chebyshev filter applied to synthetic noise trace shown in Figure 1.28.

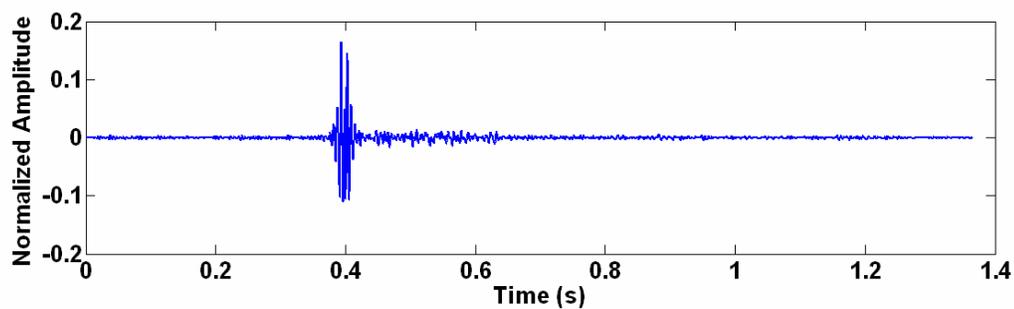


Figure 2.56: Band-pass Chebyshev filter applied to Event #1, a “good” event.

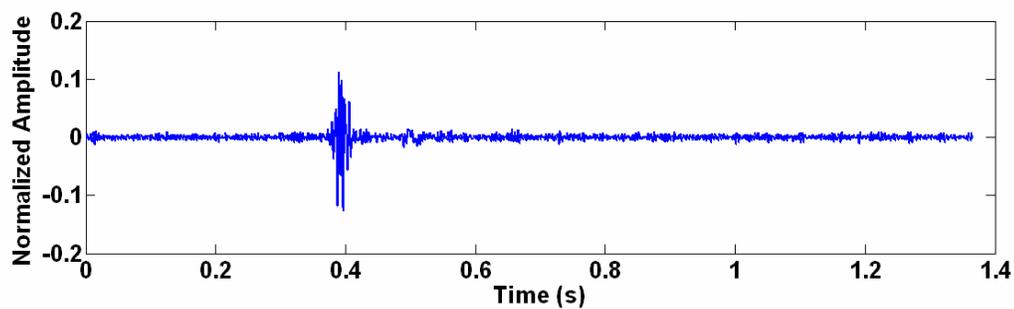


Figure 2.57: Band-pass Chebyshev filter applied to Event #2, a “good” event.

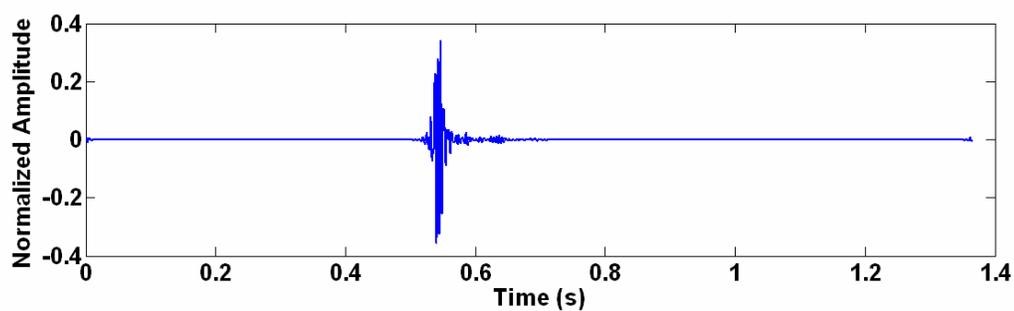


Figure 2.58: Band-pass Chebyshev filter applied to Event #3, a “good” event.

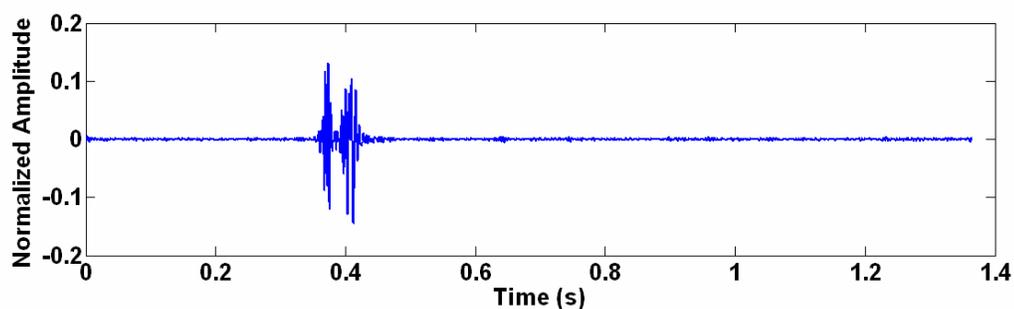


Figure 2.59: Band-pass Chebyshev filter applied to Event #4, a “good” event.

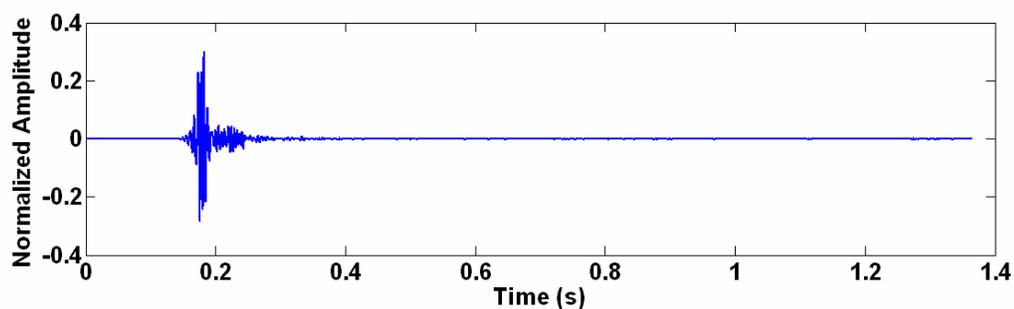


Figure 2.60: Band-pass Chebyshev filter applied to Event #5, a “good” event.

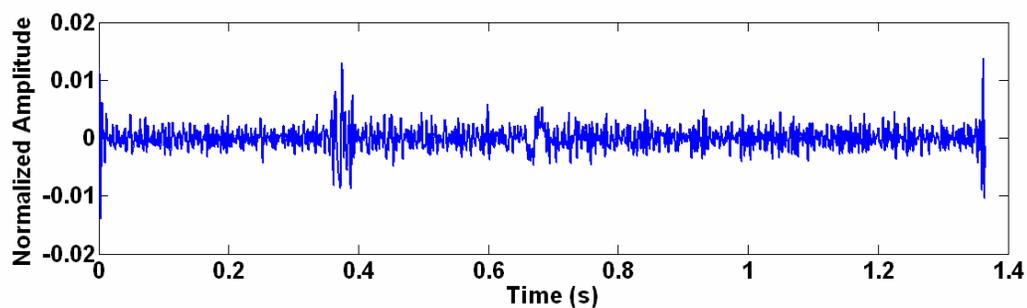


Figure 2.61: Band-pass Chebyshev filter applied to Event #6, a “good” event.

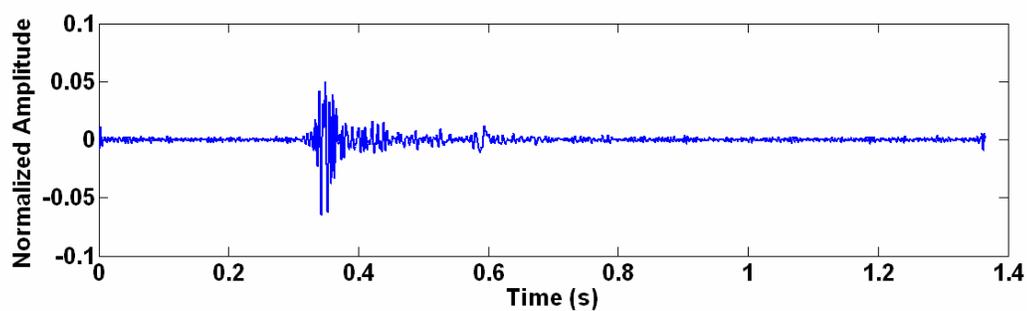


Figure 2.62: Band-pass Chebyshev filter applied to Event #7, a “good” event.

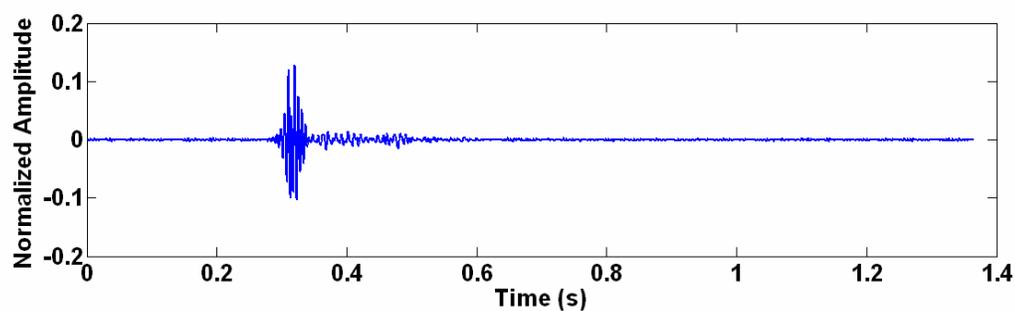


Figure 2.63: Band-pass Chebyshev filter applied to Event #8, a “good” event.

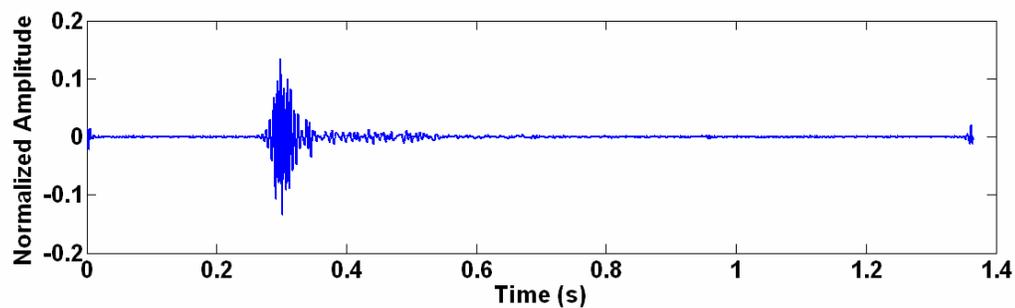


Figure 2.64: Band-pass Chebyshev filter applied to Event #9, a “good” event.

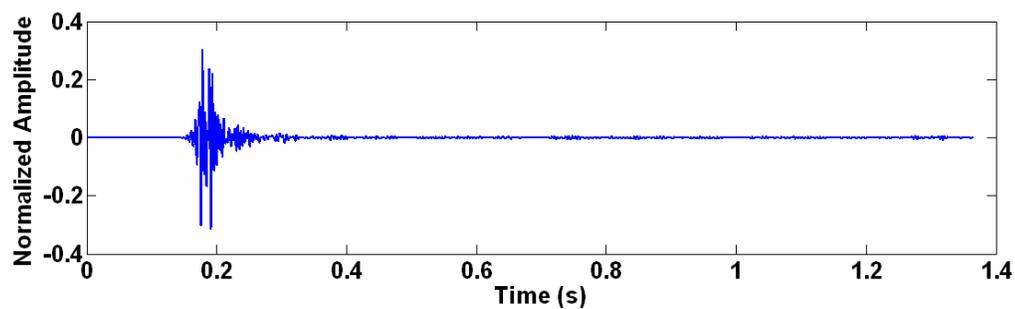


Figure 2.65: Band-pass Chebyshev filter applied to Event #10, a “good” event.

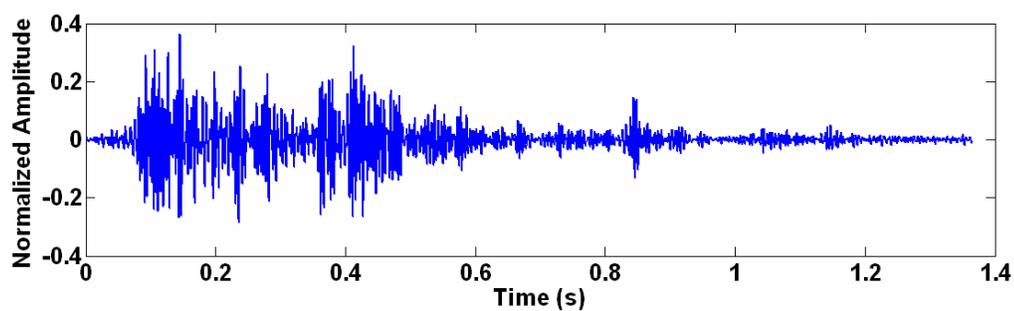


Figure 2.66: Band-pass Chebyshev filter applied to Event #11, a noise event.

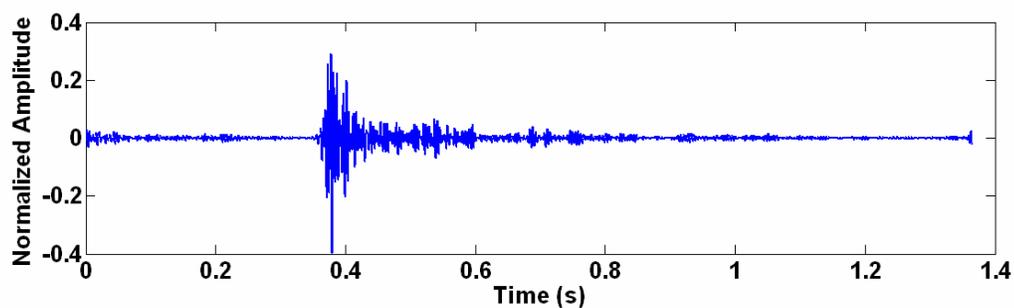


Figure 2.67: Band-pass Chebyshev filter applied to Event #12, a noise event.

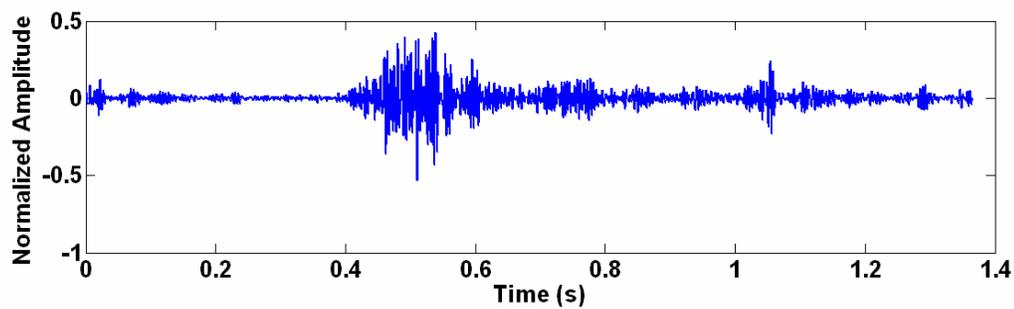


Figure 2.68: Band-pass Chebyshev filter applied to Event #13, a noise event.

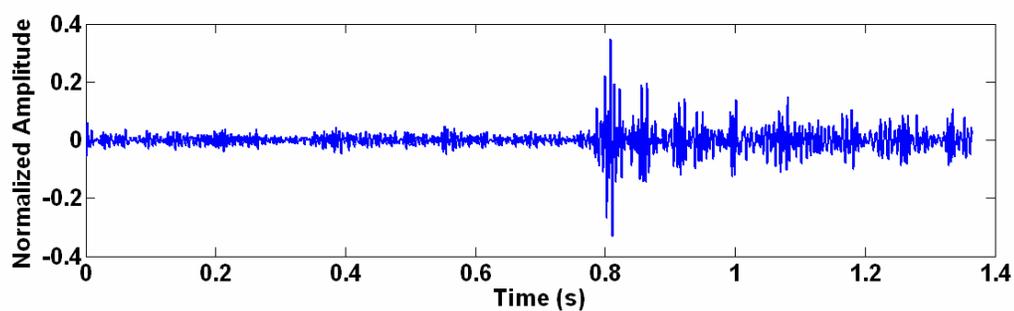


Figure 2.69: Band-pass Chebyshev filter applied to Event #14, a noise event.

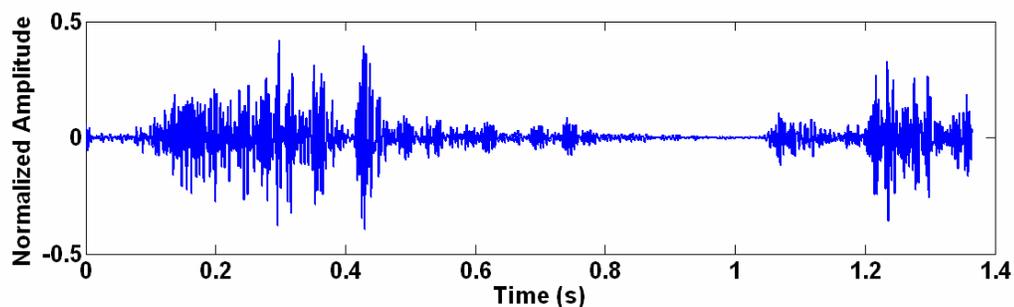


Figure 2.70: Band-pass Chebyshev filter applied to Event #15, a noise event.

## 2.6 Conclusion

I observed that many "good" events generally have lower dominant frequency content than noise events. Frequency filtering applied to microseismic signals was demonstrated to illustrate the potential of this technique regarding event classification. Background on frequency filter terminology and theory were provided. The characteristics of the Butterworth, Chebyshev, and Inverse-Chebyshev frequency response approximations were explored prior to application.

A low-pass Inverse-Chebyshev filter, a high-pass Butterworth filter, and a band-pass Chebyshev filter, all fourth order, were developed and applied to example "good" and noise traces. Filtered results showed example "good" traces having lower frequency content than noise traces, through peak-amplitude examination of the filtered traces. These results suggest that these algorithms are potentially capable of event-classification.

## **CHAPTER THREE: Event-Length Detection**

### **3.1 Introduction**

Event detection has been a major area of interest in the field of earthquake detection. When an earthquake occurs, energy is propagated outwards from its focus, resulting in induced ground motions. As more than 50,000 earthquakes and mining blasts occur annually worldwide (Withers et al., 1998), seismologists look for automatic methods to detect these events in order to estimate the location of the earthquake's hypocenter and alert the public if necessary. This type of detection is critical to ensure maximal public safety. Automatic detection of earthquakes in the presence of environmental noise has been a major focus of research in the seismological world for over 40 years (Joswig, 1990).

There is a wide range of detection algorithms that have been proposed and / or discussed by Ambuter and Solomon (1974), Engdahl et al. (1998), Withers et al. (1998), Chael (1997), Allen (1997), and Horiuchi et al. (2005), among many others. Many of these algorithms are specialized towards earthquake (as opposed to microseism) detection, but major fundamental concepts can be applied to microseisms generated during thermal oil recovery as well.

One of the most common earthquake event-detection methods that can also be applied to microseism detection concerns calculating moving averages of short-term to long-term energy ratios, also known as the STA / LTA method (Ambuter and Solomon, 1974). This technique was applied as one of the event-classification algorithms. Another event-length detection algorithm based in the frequency-domain was developed and will be discussed. These algorithms were applied to microseismic event-classification based on the

observation that the P-wave event-length in many “good” traces was usually significantly shorter than noise event-lengths.

### 3.2 STA / LTA Algorithm

This time-domain algorithm continuously calculates ratios of short-term averages (STA) to long-term averages (LTA) of microseismic energy. This is the STA/LTA technique (Ambuter and Solomon, 1974) demonstrated by Munro (2005). In microseismic analysis, the STA/LTA ratio will significantly increase at the onset of a microseismic event; conversely, this ratio will significantly decrease at the event’s termination. Calculating the time interval between the onset and termination of a microseismic event yields its approximate length.

To demonstrate an application of this algorithm to microseismic event detection, assume that there are  $N$  data samples in the microseismic trace under examination. Take the sampling frequency to be  $f_s$ . Define  $S$  as the length of the STA window in seconds containing  $p$  data points. Similarly, define  $L$  as the length of the LTA window in seconds containing  $q$  data points. Then,

$$S = \frac{p}{f_s}, \quad (3.1)$$

and

$$L = \frac{q}{f_s}. \quad (3.2)$$

Let  $a_i$  represent the amplitude of an arbitrary data point in the microseismic trace with  $i = 0, 1, 2, \dots, N-1$ . The relationship between a point in time  $t_i$  and the index “ $i$ ” is

$$t_i = \frac{i}{f_s}. \quad (3.3)$$

The following application of this algorithm was found to work best for microseismic event-length analysis. To perform the first STA / LTA calculation, the LTA window was positioned such that it calculated a long-term energy average from the beginning of the trace at  $t = 0$  to the time  $t = L$ . Using equation 3.3, this corresponds to data point indices from  $i = 0$  to  $i = Lf_s$ . The STA window was positioned such that it calculated a short-term energy average of the trace from the time  $t = L - S$  to the time  $t = L$ . This corresponds to data point indices from  $i = (L-S)f_s$  to  $i = Lf_s$ . Let  $STA_x$  and  $LTA_x$  represent the  $x^{th}$  short-term and long-term energy calculations, respectively. Let  $R_x$  signify the  $x^{th}$  calculated STA / LTA ratio. Then, the first STA and LTA calculations are given by

$$STA_1 = \frac{\sum_{i=(L-S)f_s}^{Lf_s} a_i^2}{p}, \quad (3.4)$$

and

$$LTA_1 = \frac{\sum_{i=0}^{Lf_s} a_i^2}{q}. \quad (3.5)$$

The resulting STA / LTA ratio for this first calculation is then

$$R_1 = \frac{STA_1}{LTA_1}. \quad (3.6)$$

Equations 3.4, 3.5, and 3.6 are specific to the first calculated STA / LTA ratio.

These equations can be easily modified to give the  $x^{th}$  calculated STA / LTA ratio,

assuming that the windows are advanced by single data point increments prior to subsequent calculations. The equations for the generalized quantities  $STA_x$ ,  $LTA_x$ , and  $R_x$  are

$$STA_x = \frac{\sum_{i=(L-S+x-1)f_s}^{Lf_s+x-1} a_i^2}{p}, \quad (3.7)$$

$$LTA_x = \frac{\sum_{i=x-1}^{Lf_s+x-1} a_i^2}{q}, \quad (3.8)$$

and

$$R_x = \frac{STA_x}{LTA_x}. \quad (3.9)$$

The STA / LTA algorithm applied here moves the energy windows by single data point increments until a significant increase in  $R_x$  is seen, indicating the detection of the P-wave arrival. The termination of the event is detected through a significant decrease in  $R_x$ . If no event is detected, then calculations of  $R_x$  terminate when the energy windows reach the end of the microseismic trace.

Figure 3.1 is a reproduction of Event #1 shown in chapter 1, a “good” event. The empirically picked P-wave event-length is somewhat subjective depending on how one wishes to pick the exact onset and termination points of the event. Here, it is empirically picked to have an event-length of 31 ms.

Figures 3.2 to 3.7 show the calculated STA / LTA ratios as functions of the STA window position. For example, a time point of 200 ms on this axis corresponds to the STA window having limits from  $t = 200$  ms to  $t = 200 + S$  ms for that specific STA / LTA calculation. The temporal window length  $L$  of the LTA window is set to  $L = 100$  ms. The temporal length of the STA window is varied from 2 ms to 50 ms in these figures to illustrate the effect of varying this parameter. This window length  $S$  must be carefully chosen in a manner such that it is not too long to ensure that short term energy trends can be distinguished from long term energy trends, but  $S$  must also not be too short to ensure that noise does not drive the calculated values of  $R_x$ . A reasonably accurate approximation of the P-wave event length is also desired.

The onset of an event can be detected where there is a sharp increase in  $R_x$ , and the termination of an event can be similarly detected where there is a sharp decrease in  $R_x$ . These sharp changes can be seen at around 400 ms. By examining Figures 3.2 to 3.7, it is clear that, for this example, if  $S$  is too small, the calculated event-length will be too short; conversely, if  $S$  is too large, the calculated event-length will be too long. A value of  $S = 20$  ms gives the most accurate pick for the example in Figure 3.1. Table 3.1 provides a summary of the detected event-lengths using this STA / LTA technique. Note that the event-lengths of these example “good” events are considerably shorter than those of the example noise events, suggesting that this algorithm is capable of event-classification.

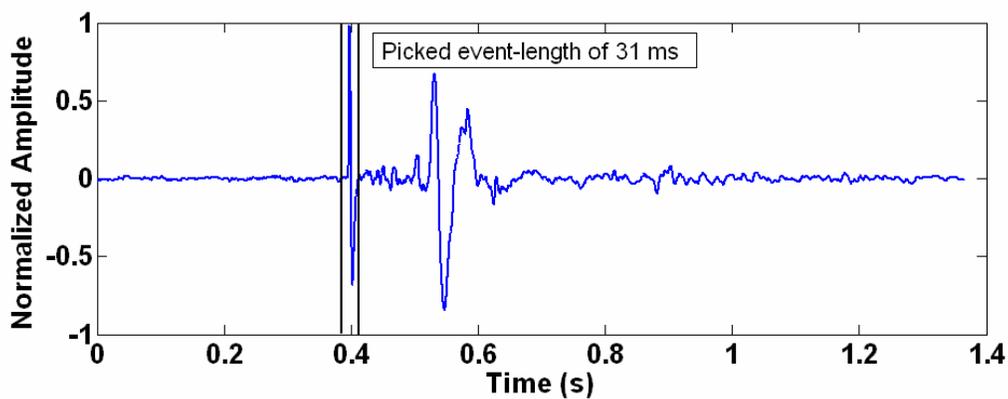


Figure 3.1: A “good” event with an empirically picked P-wave event-length of 31 ms.

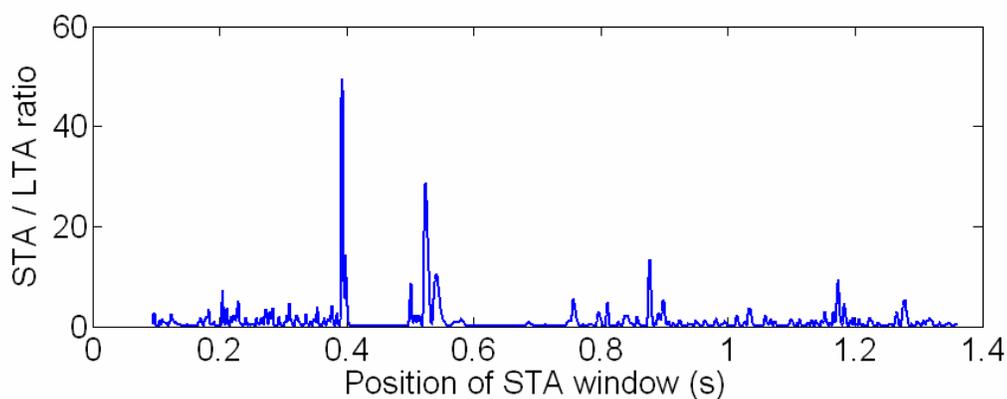


Figure 3.2: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 2$  ms. The automatically picked event-length is 3.6 ms.

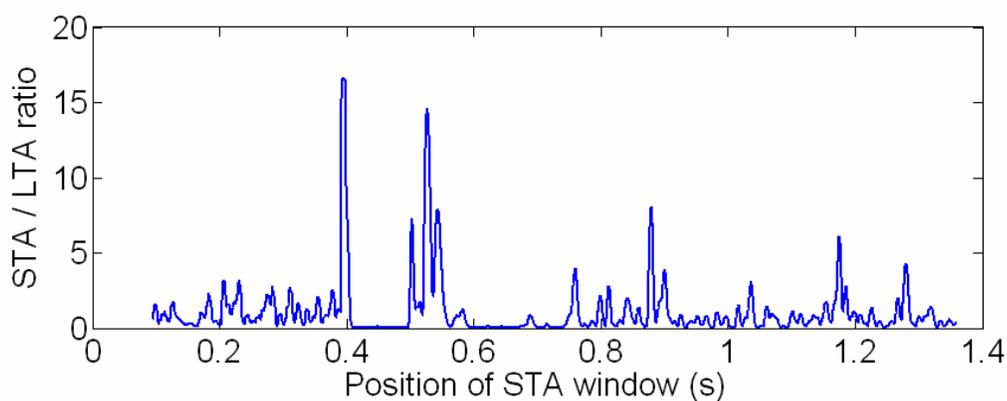


Figure 3.3: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 6$  ms. The automatically picked event-length is 9.1 ms.

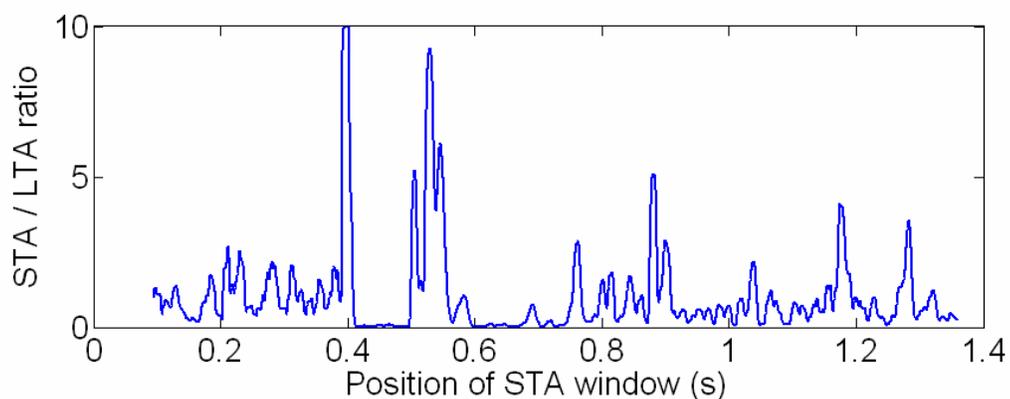


Figure 3.4: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 10$  ms. The automatically picked event-length is 12.6 ms.

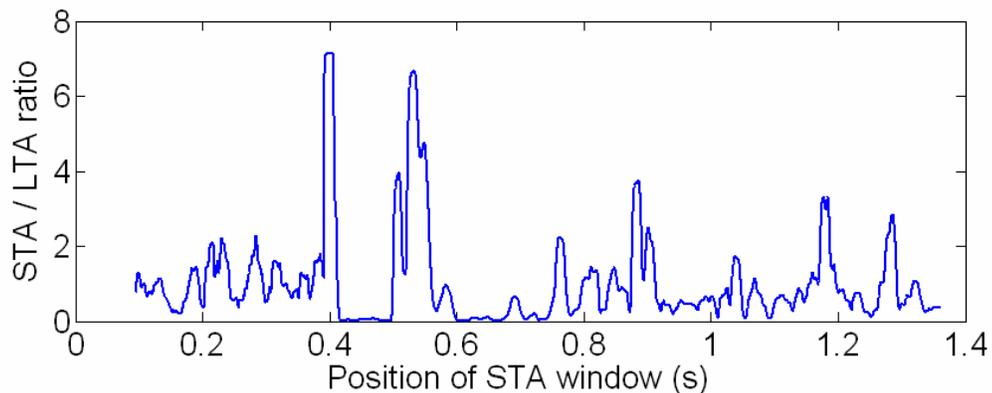


Figure 3.5: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 14$  ms. The automatically picked event-length is 17.2 ms.

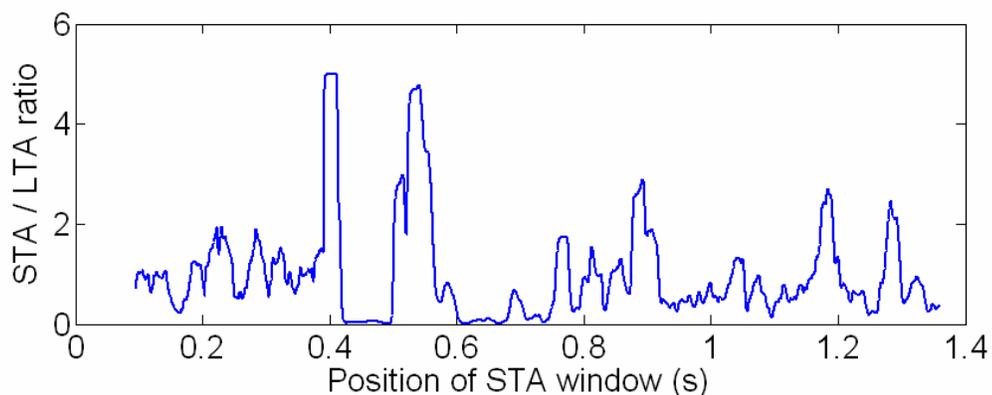


Figure 3.6: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 20$  ms. The automatically picked event-length is 30.3 ms.

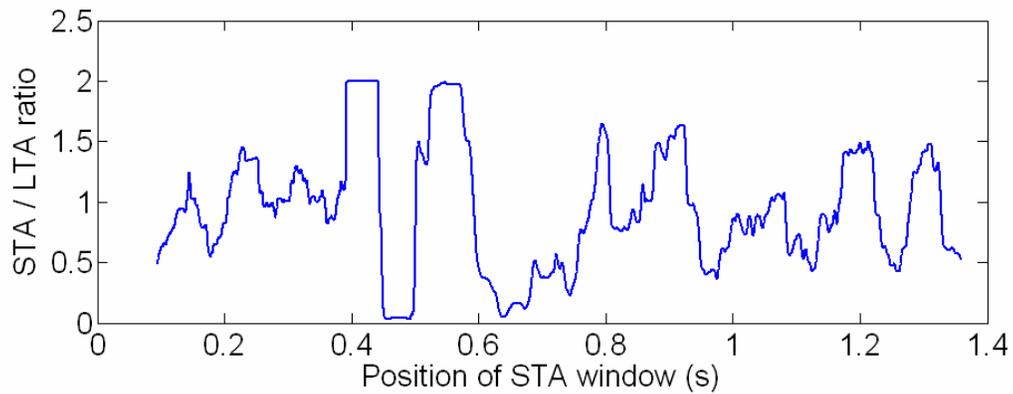


Figure 3.7: Calculated STA / LTA ratios of the trace in Figure 3.1 with  $S = 50$  ms. The automatically picked event-length is 52.4 ms.

Table 3.1: Summary of detected event-lengths using STA / LTA technique. Note that event-lengths of example “good” events are considerably shorter than those of example noise events.

Event Type	Event #	Event-Length (ms)
Good	1	30
Good	2	28
Good	3	59
Good	4	41
Good	5	36
Good	6	31
Good	7	39
Good	8	40
Good	9	38
Good	10	51
Noise	11	531
Noise	12	102
Noise	13	197
Noise	14	528
Noise	15	379

### 3.3 Continuous-Time Frequency Analysis

It was observed that the high-frequency content of many microseismic traces usually increased significantly at the onset of an event and decreased significantly at the event's termination.

A developed method to pick the onset and termination times of microseismic events was to continually analyze the frequency characteristics of a select number of points in the channel. Defining a time window from times  $t_1$  to  $t_2$  in the channel, a continuous-time frequency analysis was performed by supplying a moving time window that existed between  $t_1$  and  $t_2$ , with the limits  $t_1$  and  $t_2$  continuously increased up until the end of the channel data. This developed technique is similar to the Gabor transform (e.g. Feichtinger and Strohmer, 1998) and S-transform (Stockwell et al., 1996) techniques which represent frequency transform methods that can be localized in time. These transform methods are useful when dealing with signals that are not stationary. A stationary signal (or wavelet) is one whose waveform does not change shape over time (e.g. Margrave, 2007). The time series in an earthquake seismogram is not stationary (e.g. Stockwell et al., 1996). Thus, when analyzing microseismic signals, especially when event-lengths and temporal locations are desired, it is preferred to use a continuous-time frequency transform technique capable of temporally localizing signal frequency characteristics.

For a continuous signal, frequency characteristics are determined using the Fourier transform, defined as

$$G(f) = \int_{-\infty}^{\infty} g(t) \exp(-j2\pi ft) dt \quad (3.10)$$

(e.g. Haykin, 2001). In applying equation 3.10, the function  $g(t)$  represents the data in the microseismic channel between the intervals  $t_1$  and  $t_2$ , explained above. Thus, as  $t_1$  and  $t_2$  are continuously increased, a continuous-time Fourier transform is performed. The power spectral density (PSD),  $|G(f)|^2 = G(f)G^*(f)$ , can be obtained where “\*” is the complex conjugate operator. The procedure can be repeated as the limits  $t_1$  and  $t_2$  are increased. The obtained result can be examined to determine onset and termination times of microseismic events.

For discrete signals, the discrete Fourier transform (DFT) is used to determine frequency characteristics. The generalized definition of the DFT is

$$f_m = \sum_{n=0}^{N-1} f_n e^{\frac{-j2\pi mn}{N}} \quad (3.11)$$

(e.g. Margrave, 2007). In equation 3.11,  $n$  corresponds to the indices of the time-series elements  $f_n$  (with  $n = 0, 1, 2, 3 \dots N-1$ ),  $m$  corresponds to the indices of the frequency elements  $f_m$  (with  $m = 0, 1, 2, 3 \dots N-1$ ), and  $N$  is the number of array elements.

Let an examined trace sampled at a frequency  $f_s$  contain  $N$  data points. Let  $d_i$  represent an arbitrary data point in this trace with  $i = 1, 2, 3 \dots N$ . The time value  $t$  corresponding to an index  $i$  is

$$t = \frac{i}{f_s}. \quad (3.12)$$

Let  $W$  represent a fast Fourier transform (FFT) boxcar window containing  $k$  data points. Define  $W_p$  as the  $p^{th}$  data point contained in  $W$ , with  $p = 1, 2, 3 \dots k$ . The fast Fourier transform (FFT) is a built-in MATLAB<sup>®</sup> function that calculates the discrete Fourier transform using an optimized method to reduce computation time.

To start,  $W$  is positioned at the beginning of the data, and an FFT of data points from  $d_1$  to  $d_k$  is applied. The  $k$  time-series elements are transformed to  $k$  frequency elements. Let  $f_m$  represent the  $m^{\text{th}}$  frequency element of the transformed data (with  $m = 1, 2, 3 \dots k$ ) contained in the window  $W$ . For this algorithm, equation 3.11 can be rewritten as

$$f_m = \sum_{p=1}^k W_p e^{\frac{-j2\pi(m-1)(p-1)}{k}} . \quad (3.13)$$

The Nyquist frequency  $f_{NYQ}$  is the highest signal frequency possible in a sampling process before aliasing becomes present in the reconstructed signal (e.g. Sheriff and Geldart, 1995) and is given as

$$f_{NYQ} = \frac{f_s}{2} . \quad (3.14)$$

Let  $F_m$  represent the frequency (in Hz) that each corresponding transformed frequency element  $f_m$  pertains to. The expression for  $F_m$  will depend on whether  $k$  is odd or even. If  $k$  is even, the equation is

$$F_m = \frac{(m-1)f_s}{k} , \quad m \leq \frac{k}{2} + 1; \quad (3.15a)$$

$$F_m = \frac{(m-1)f_s}{k} - 2f_{NYQ}, \quad m > \frac{k}{2} + 1, \quad (3.15b)$$

whereas if  $k$  is odd, the expression is

$$F_m = \frac{(m-1)f_s}{k} , \quad m \leq \frac{k-1}{2} + 1; \quad (3.16a)$$

$$F_m = \frac{(m-1)f_s}{k} - 2f_{NYQ}, \quad m > \frac{k-1}{2} + 1. \quad (3.16b)$$

Equations 3.15a and 3.16a correspond to positive frequencies, while equations 3.15b and 3.16b correspond to negative frequencies. Comparing these equations with equation 3.14, note that  $|F_m| \leq f_{NYQ}$  for all  $m$ .

After this FFT step is complete, the PSD of the frequency elements  $f_m$  is calculated. Let  $P_m$  represent the PSD magnitude corresponding to a transformed frequency element  $f_m$  given by

$$P_m = f_m f_m^* = |f_m|^2. \quad (3.17)$$

If the time series contained in  $W$  is real, as expected for a microseismic trace, then it can be shown that the amplitude spectrum of the transformed elements,  $|f_m|$ , is an even function in frequency (e.g. Haykin and Van Veen, 2003), implying that  $P_m$  is an even function in frequency as well. Thus, transformed elements corresponding to negative frequencies can effectively be ignored, as they are redundant, and this analysis can be restricted to positive frequencies to reduce computation time. In other words,  $P_m$  need only be found for  $m \leq \frac{k}{2} + 1$  (if  $k$  is even), or for  $m \leq \frac{k-1}{2} + 1$  (if  $k$  is odd). The elements in  $P_m$  are then placed in a single column of a 2-D matrix whose rows and columns correspond to frequency and time, respectively.

The above procedure is repeated as the window  $W$  moves forward by single data point increments until the end point of this window reaches  $d_N$ . There is a direct tradeoff between frequency and time resolution as the size of  $W$  (denoted by  $k$ ) changes. If  $k$  is too small, the final plot will have good time resolution, but poor frequency resolution; conversely, if  $k$  is too large, the plot will have good frequency resolution, but poor time resolution. This tradeoff is related to the Heisenberg uncertainty principle (Heisenberg,

1927), later translated to English in a textbook by Wheeler and Zurek (1983). In quantum physics, the Heisenberg uncertainty principle states that there is a tradeoff in measurement accuracy between position and momentum. Somewhat related to this is the fact that, in the continuous-time frequency analysis algorithm developed, there is a direct tradeoff between frequency and time resolution. This tradeoff will be demonstrated.

Figure 3.8 depicts a general block diagram of this algorithm. The “Matrix Reversal” operation corresponds to reversing the element order of the matrix (the positioning of the first and last matrix elements are switched, and so on). High frequency content is examined in the resulting 2-D matrix to determine the onset and termination times of an event. In Figure 3.8, the time axis is defined as the temporal position of the first data element in the time-localized window  $W$ .

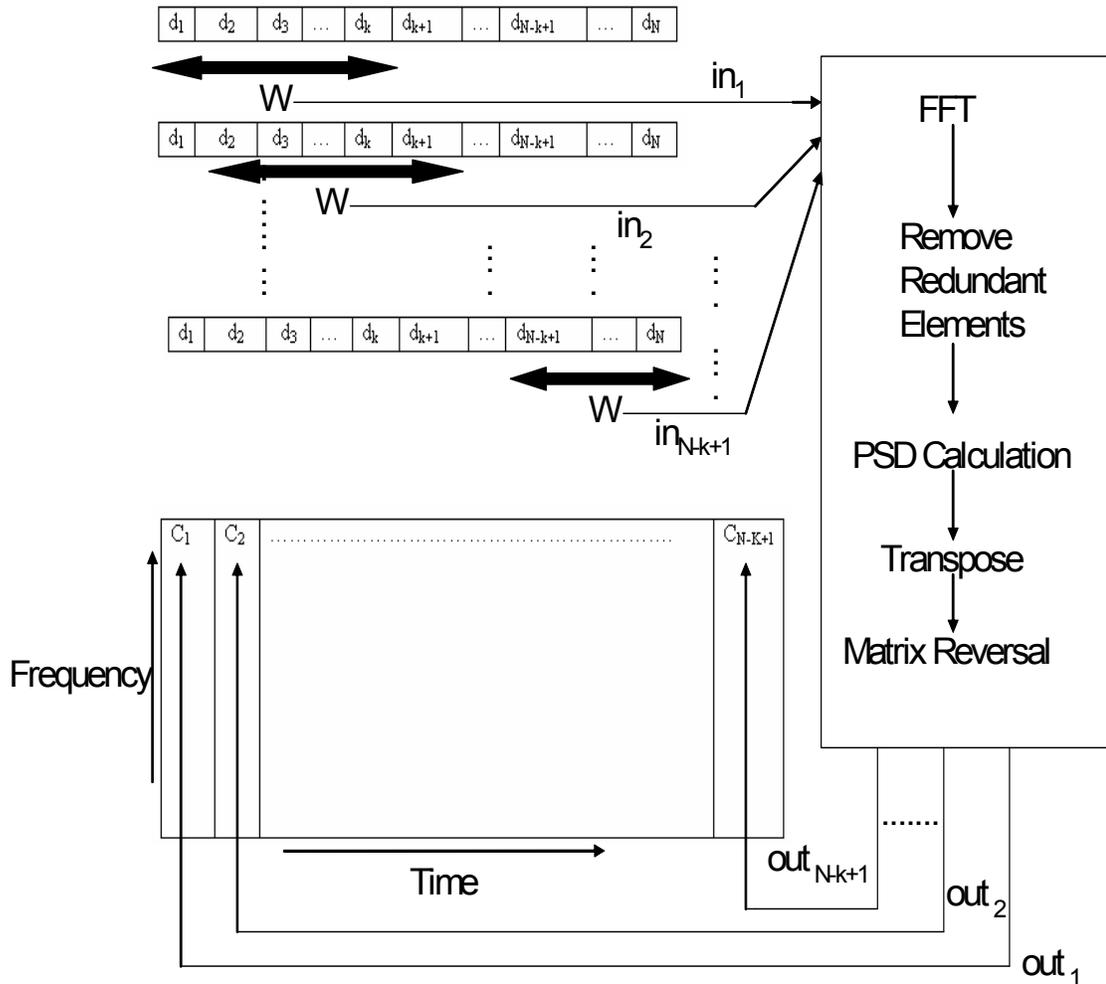


Figure 3.8: General block diagram of the developed continuous-time frequency-domain event-length detection algorithm. The row-vector represents input time data elements  $d_i$ . The output column-vectors are  $out_1, out_2$ , etc. The columns in the 2-D matrix are denoted by  $C_1, C_2$ , etc.

Consider this algorithm applied to the microseismic traces in Figures 1.7 and 1.17, which correspond to Event #1 and Event #11, respectively. Event #1 is an example of a “good” event, while Event #11 is an example of a noise event. These traces have  $N = 4096$  and  $f_s = 3000$  Hz. To demonstrate the importance of choosing an optimal size  $k$  of the FFT

window  $W$ , the frequency-time resolution tradeoff is shown in subsequent figures.

Figures 3.9 and 3.10 display Event #1 and Event #11, respectively, with this continuous-time Fourier transform algorithm applied using a very small window size ( $k = 10$ ). These figures show strong time resolution, but poor frequency resolution. Figures 3.11 and 3.12 show these transforms using a very large window size ( $k = 1000$ ). As expected, these figures have strong frequency resolution, but poor time resolution.

After experimentation, it was found that a window size of  $k = 100$  was optimal to obtain the best time-frequency resolution combination. Figure 3.13 and Figure 3.14 show the transformed “good” (Event #1) and noise (Event #11) examples, respectively. The frequency axis has an upper display limit of 250 Hz in Figure 3.13, although the resultant 2-D matrix contains elements that correspond to frequencies up to 1500 Hz, the Nyquist frequency. The frequency content was negligible above 500 Hz in Figure 3.13. The calculated P-wave event-length in Figure 3.13 using this algorithm was 35 ms. The calculated noise event-length in Figure 3.14 was 547 ms. The onset and termination event locations are also shown in these figures.

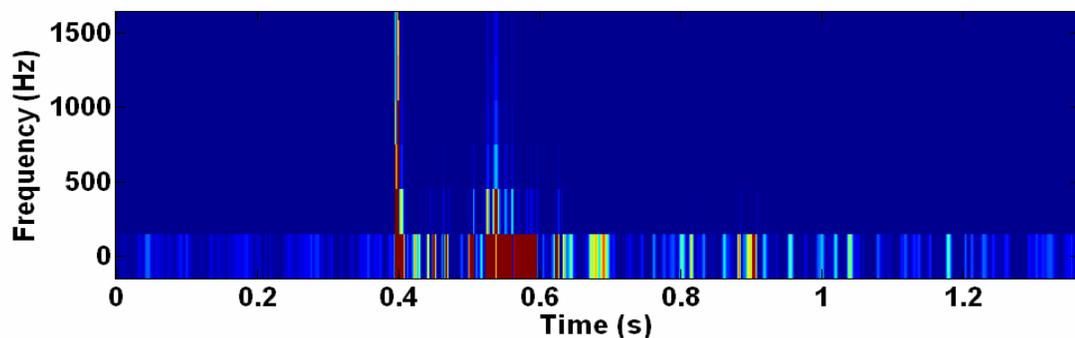


Figure 3.9: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an FFT window size  $k = 10$ . There is strong time resolution, but poor frequency resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).

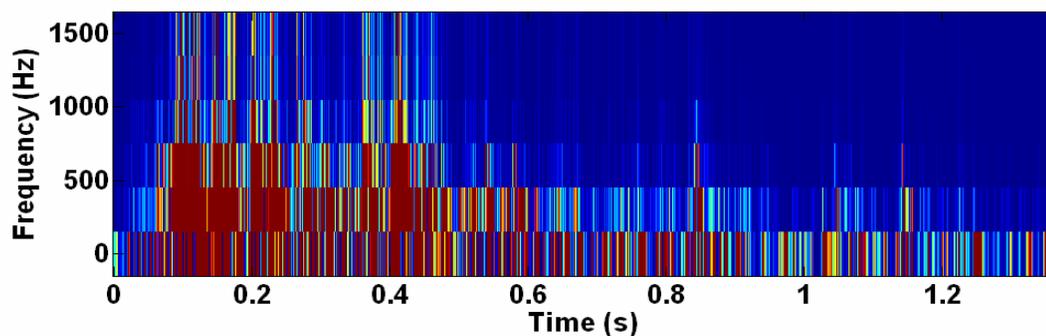


Figure 3.10: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an FFT window size  $k = 10$ . There is strong time resolution, but poor frequency resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).

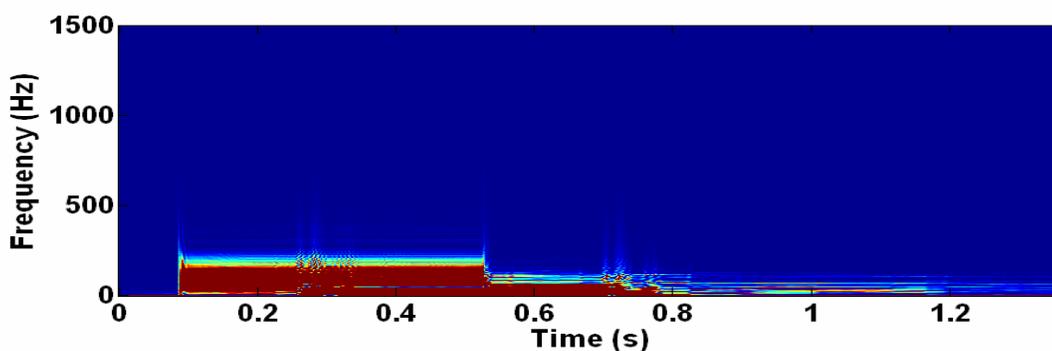


Figure 3.11: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an FFT window size  $k = 1000$ . There is strong frequency resolution, but poor time resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).

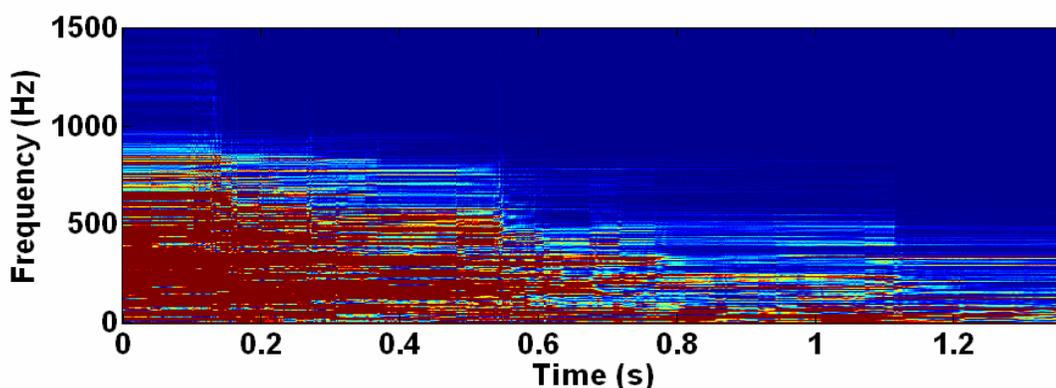


Figure 3.12: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an FFT window size  $k = 1000$ . There is strong frequency resolution, but poor time resolution. The color spectrum spans from dark blue (low-magnitude) to bright red (high-magnitude).

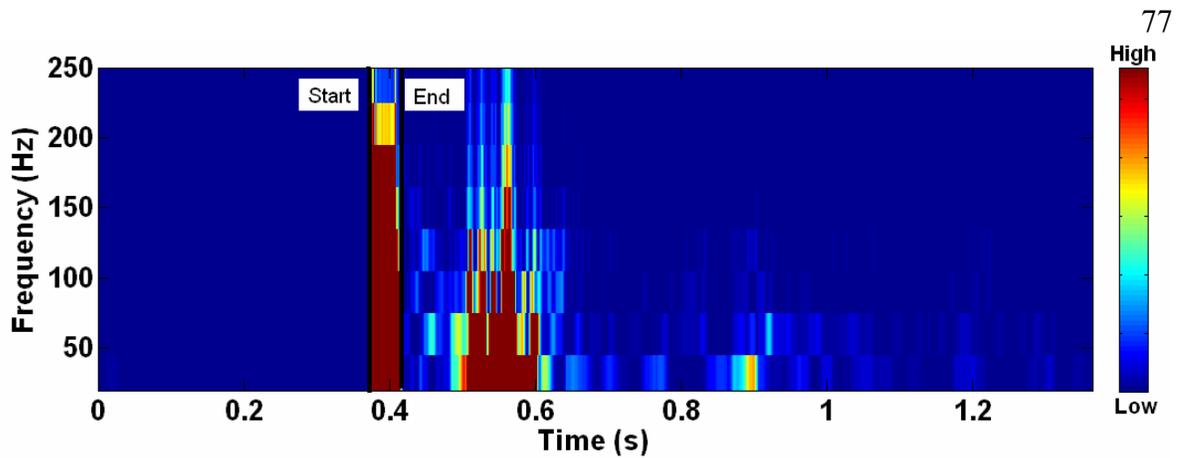


Figure 3.13: Continuous-time Fourier transform PSD plot of Event #1, a “good” trace, using an experimentally determined optimal FFT window size  $k = 100$ . There is adequately strong time and frequency resolution. The event-length was calculated to be 35 ms.

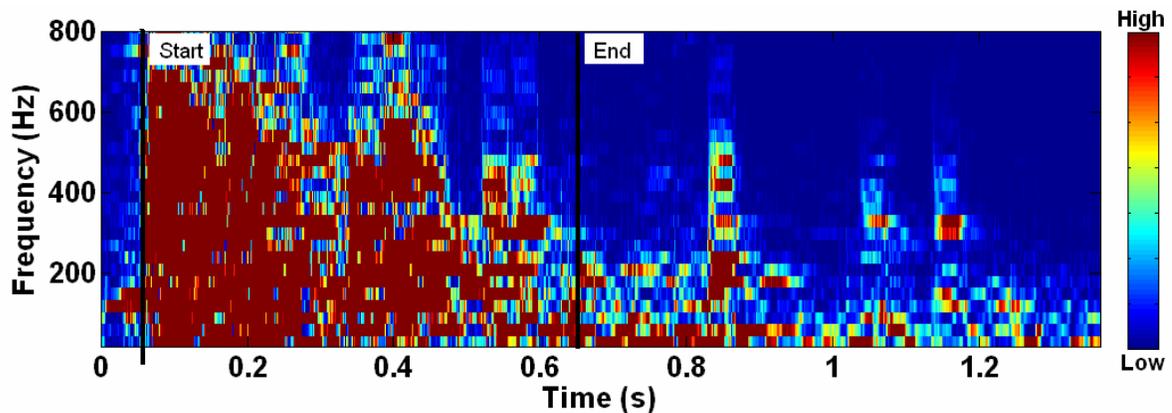


Figure 3.14: Continuous-time Fourier transform PSD plot of Event #11, a noise trace, using an experimentally determined optimal FFT window size  $k = 100$ . There is adequately strong time and frequency resolution. The event-length was calculated to be 547 ms.

Applying the optimal FFT window size of  $k = 100$ , Figures 3.15 to 3.20 show PSD plots of the synthetic “good” traces depicted in Figures 1.22 to 1.27. Although the P-wave becomes harder to distinguish as SNR decreases due to noise frequency components causing increased distortion, this arrival remains relatively easy to pick for SNR values of 15 dB or higher. The picked event-length of this P-wave arrival is 30 ms. When the SNR

decreases to 10 dB or lower, however, the PSD becomes driven by noise frequencies resulting in the P-wave arrival becoming very difficult to distinguish. In Figure 3.20, where the time-domain trace has an SNR of 0 dB, the P-wave arrival is indistinguishable in the PSD plot. Figure 3.21 shows a PSD plot of the synthetic noise trace depicted in Figure 1.28. The picked noise event length is 1.1 seconds, significantly larger than the picked “good” event lengths of 30 ms.

Figures 3.22 to 3.30 show PSD plots of Event #2 to Event #10, which are “good” events. Figures 3.31 to 3.34 show PSD plots of Event #12 to Event #15, which are noise events. Table 3.2 provides a summary of the detected event-lengths using this frequency transform technique. Note that the event-lengths of these example “good” events are considerably shorter than those of the example noise events, suggesting that this algorithm is capable of event-classification.

Table 3.2: Summary of detected event-lengths using frequency transform technique. Note that event-lengths of example “good” events are considerably shorter than those of example noise events.

<b>Event Type</b>	<b>Event #</b>	<b>Event-Length (ms)</b>
Good	1	35
Good	2	32
Good	3	62
Good	4	43
Good	5	37
Good	6	34
Good	7	45
Good	8	44
Good	9	47
Good	10	54
Noise	11	547
Noise	12	114
Noise	13	234
Noise	14	570
Noise	15	415

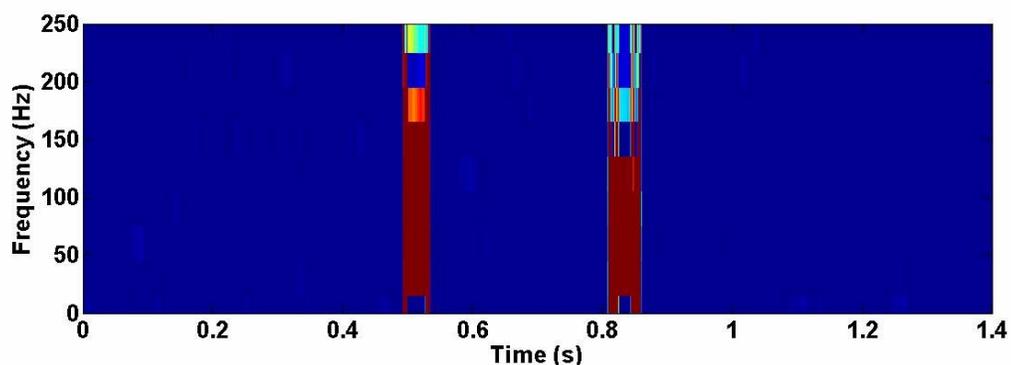


Figure 3.15: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.22 (SNR = 30 dB) with FFT window size  $k = 100$ . The P-wave event-length was calculated to be 30 ms.

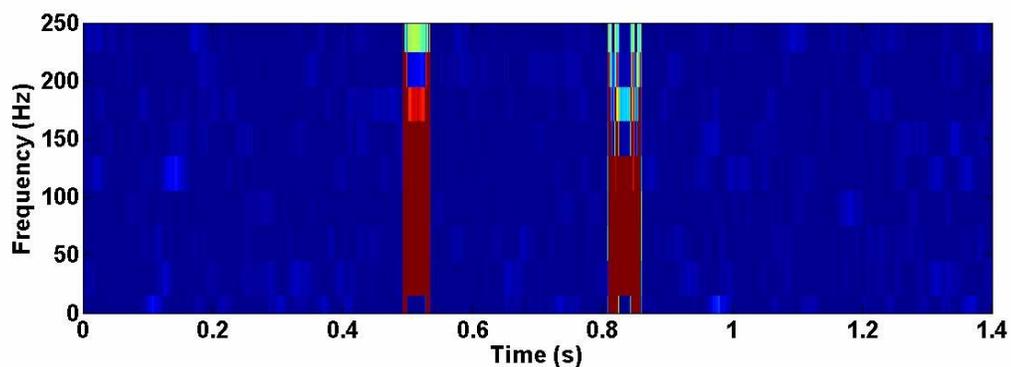


Figure 3.16: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.23 (SNR = 25 dB) with FFT window size  $k = 100$ . The P-wave event-length was calculated to be 30 ms.

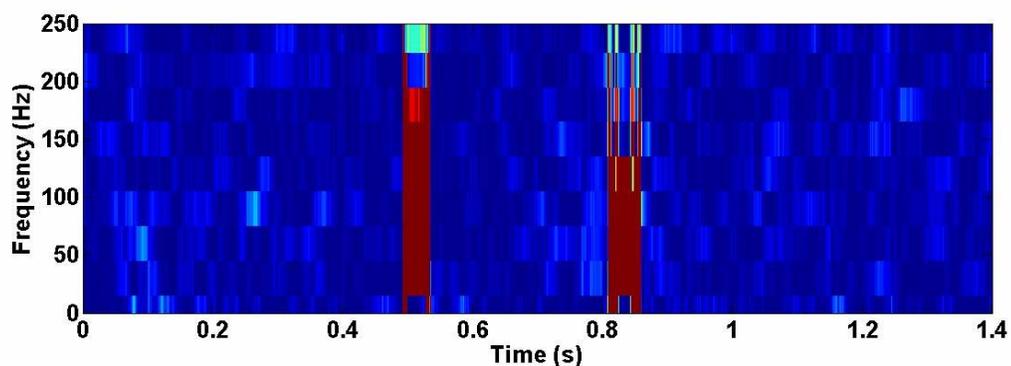


Figure 3.17: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.24 (SNR = 20 dB) with FFT window size  $k = 100$ . The P-wave event-length was calculated to be 30 ms.

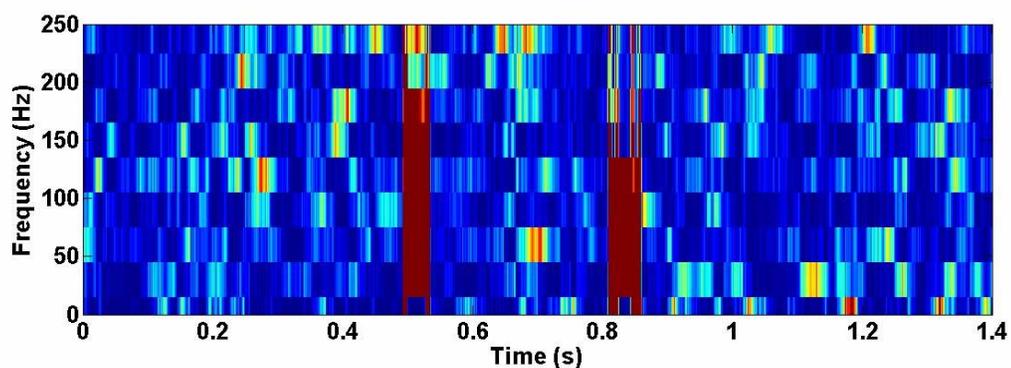


Figure 3.18: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.25 (SNR = 15 dB) with FFT window size  $k = 100$ . The P-wave event-length was calculated to be 30 ms.

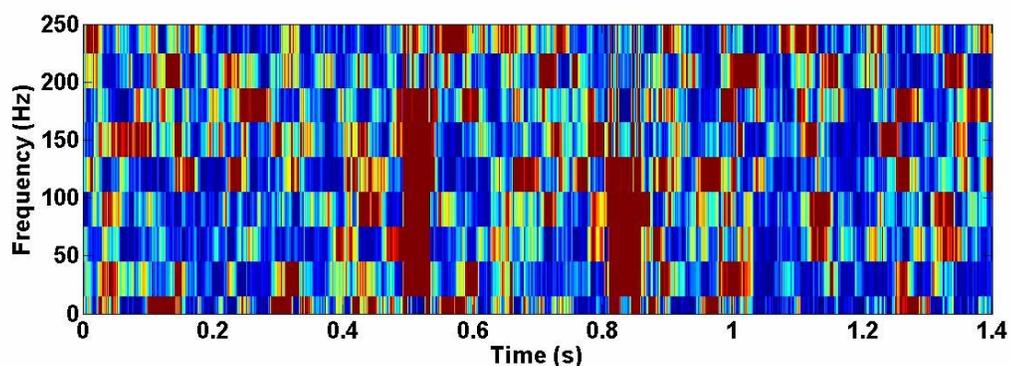


Figure 3.19: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.26 (SNR = 10 dB) with FFT window size  $k = 100$ . The P-wave event-length was calculated to be 30 ms, but significant noise frequency components increase the difficulty of distinguishing this arrival.

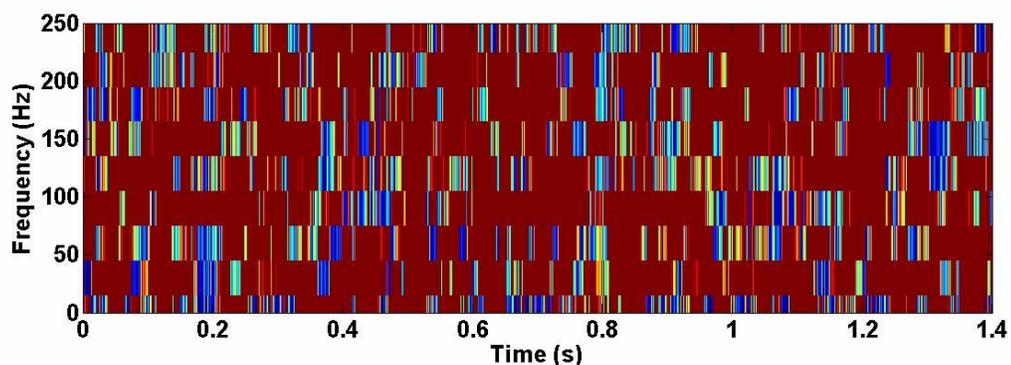


Figure 3.20: Continuous-time Fourier transform PSD plot of synthetic “good” trace in Figure 1.27 (SNR = 0 dB) with FFT window size  $k = 100$ . The P-wave arrival is indistinguishable due to very large noise frequency components.

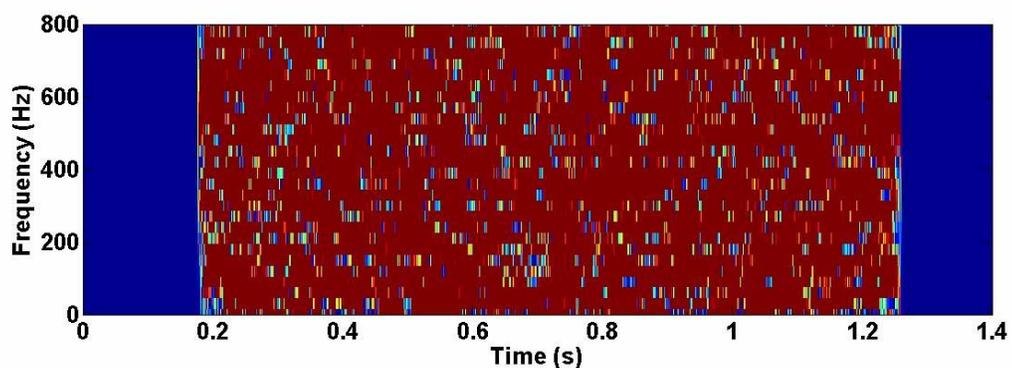


Figure 3.21: Continuous-time Fourier transform PSD plot of synthetic noise trace in Figure 1.28 with FFT window size  $k = 100$ . The calculated event-length was 1.1 seconds.

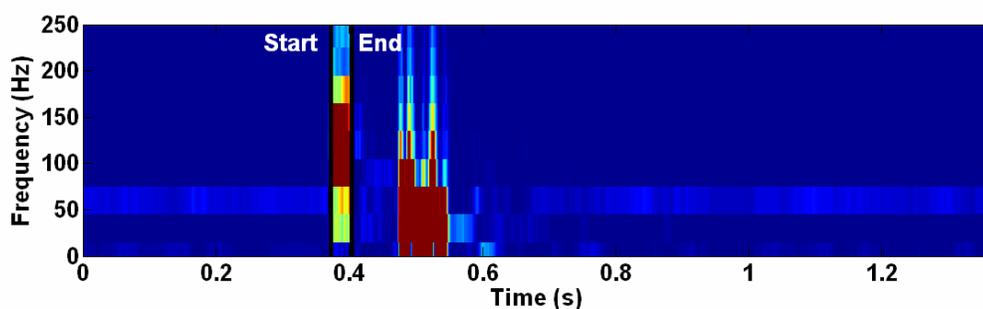


Figure 3.22: Continuous-time Fourier transform PSD plot of Event #2, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 32 ms.

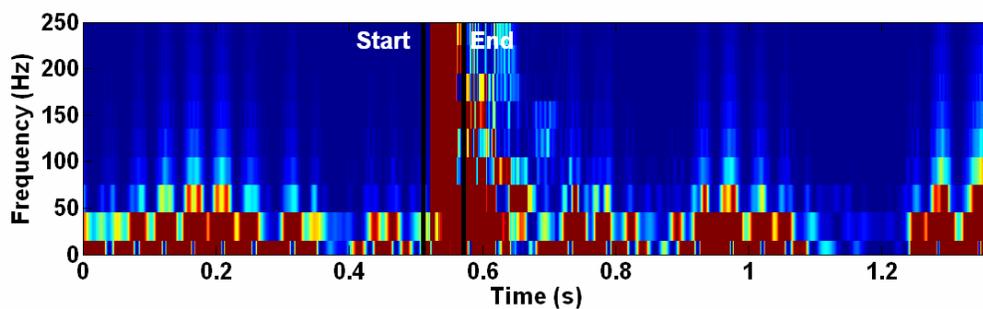


Figure 3.23: Continuous-time Fourier transform PSD plot of Event #3, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 62 ms.

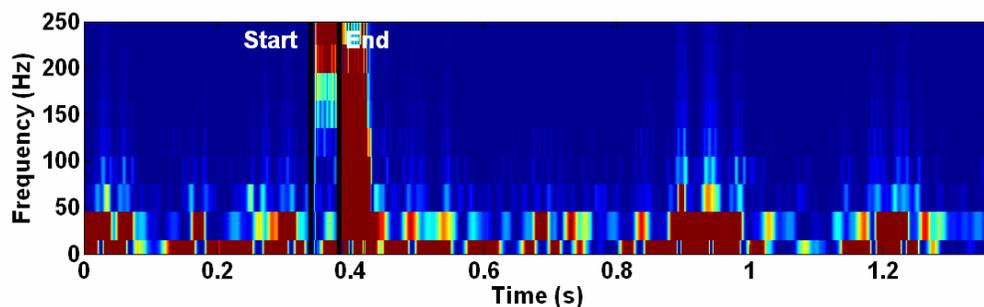


Figure 3.24: Continuous-time Fourier transform PSD plot of Event #4, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 43 ms.

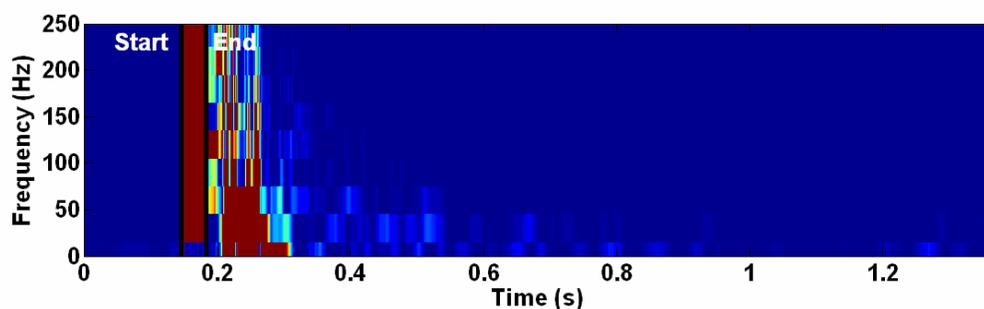


Figure 3.25: Continuous-time Fourier transform PSD plot of Event #5, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 37 ms.

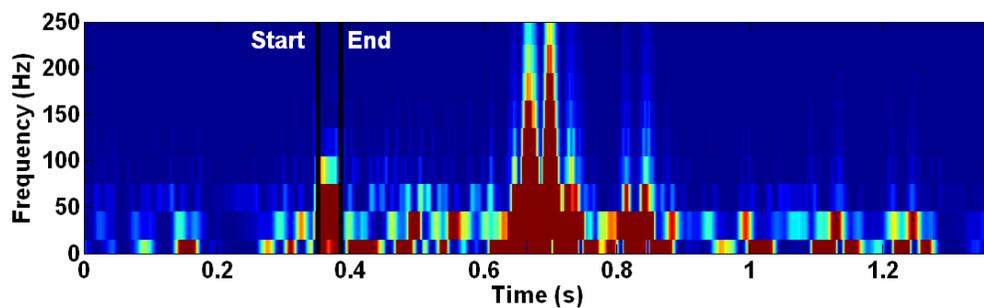


Figure 3.26: Continuous-time Fourier transform PSD plot of Event #6, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 34 ms.

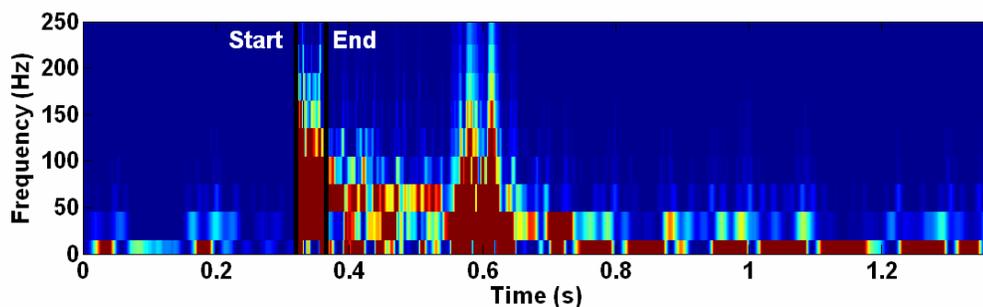


Figure 3.27: Continuous-time Fourier transform PSD plot of Event #7, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 45 ms.

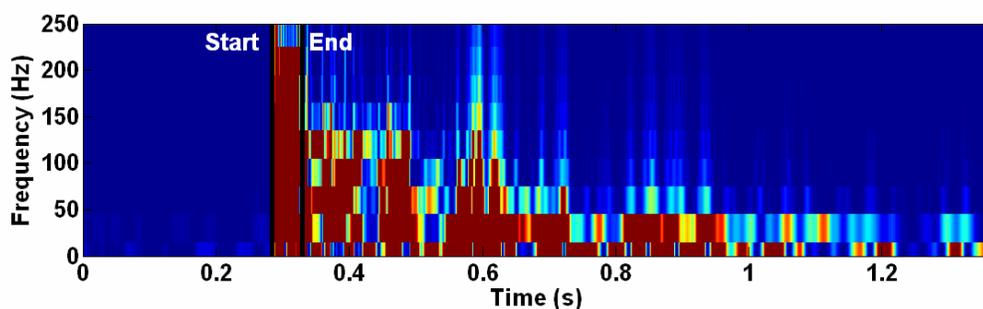


Figure 3.28: Continuous-time Fourier transform PSD plot of Event #8, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 44 ms.

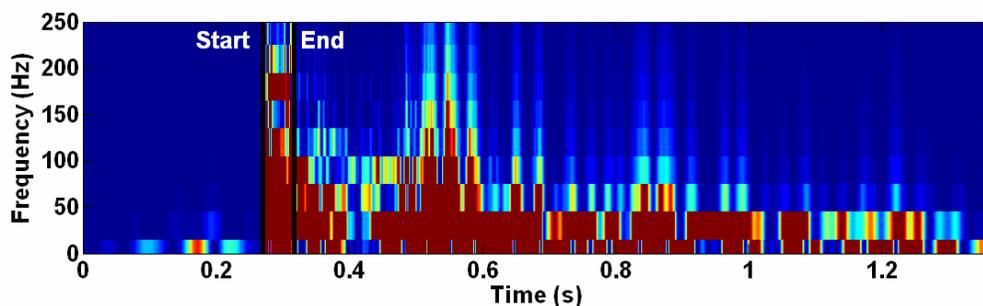


Figure 3.29: Continuous-time Fourier transform PSD plot of Event #9, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 47 ms.

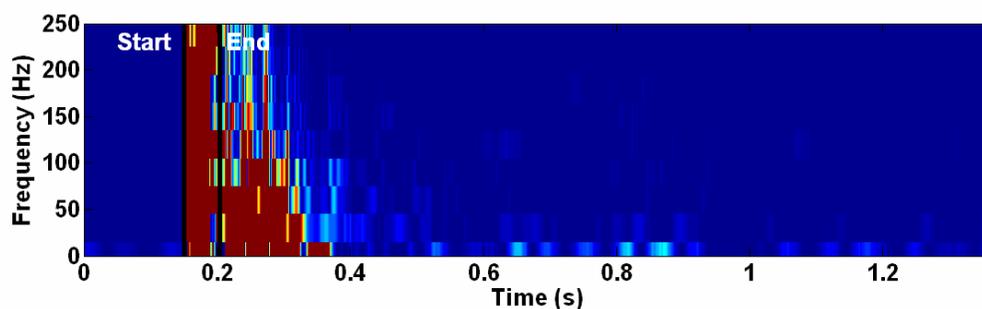


Figure 3.30: Continuous-time Fourier transform PSD plot of Event #10, a “good” event, with FFT window size  $k = 100$ . The calculated event-length was 54 ms.

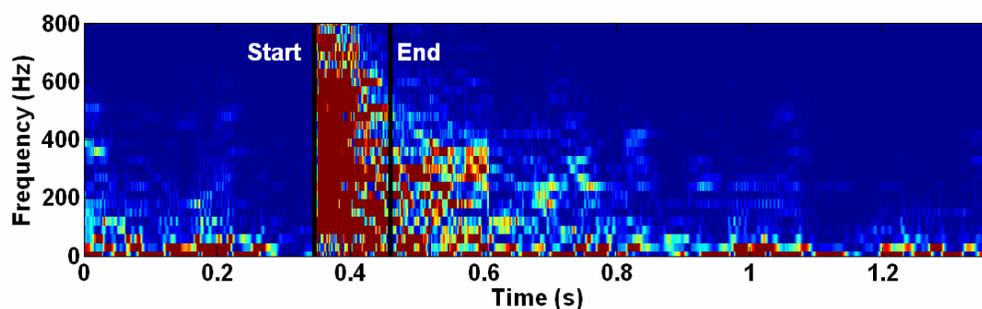


Figure 3.31: Continuous-time Fourier transform PSD plot of Event #12, a noise event, with FFT window size  $k = 100$ . The calculated event-length was 114 ms.

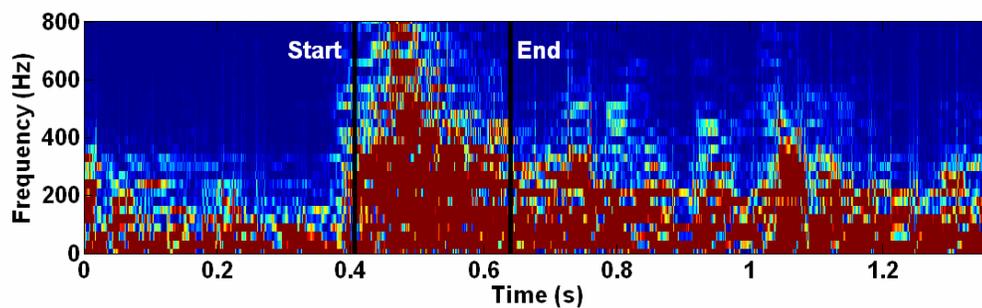


Figure 3.32: Continuous-time Fourier transform PSD plot of Event #13, a noise event, with FFT window size  $k = 100$ . The calculated event-length was 234 ms.

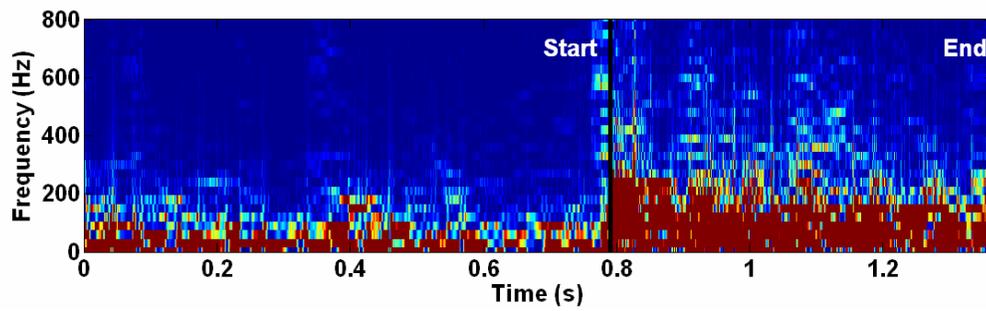


Figure 3.33: Continuous-time Fourier transform PSD plot of Event #14, a noise event, with FFT window size  $k = 100$ . The calculated event-length was 570 ms.

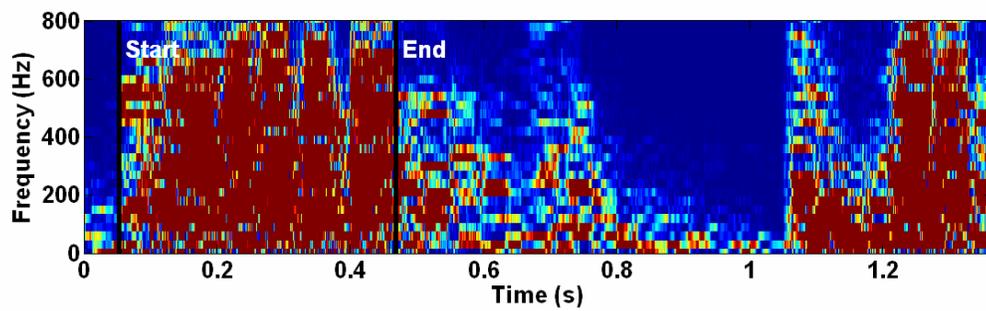


Figure 3.34: Continuous-time Fourier transform PSD plot of Event #15, a noise event, with FFT window size  $k = 100$ . The calculated event-length was 415 ms.

### 3.4 Conclusion

I observed that the P-wave event-lengths in many “good” traces are usually significantly shorter than noise event-lengths. Thus, two event-length detection techniques, one in the time domain and the other in the frequency domain, were performed to differentiate “good” microseismic signals from noise based on calculated event-lengths.

The time domain technique continuously calculated ratios of short-term averages (STA) to long-term averages (LTA) of microseismic energy to determine event onset and termination times. This is known in literature as the STA/LTA technique. The developed frequency domain technique continuously analyzed the frequency characteristics of a select number of points in the channel through a moving time window, similar to the Gabor and S-transform techniques, and examined high frequency content to determine event onset and termination times.

The effects of varying window sizes pertaining to both the time and frequency domain techniques were examined. After empirically optimizing these parameters, these techniques appear capable of event-length detection. For example, the calculated noise trace event-lengths, shown in Figures 3.31 to 3.34, were significantly longer than the “good” event-lengths, shown in Figures 3.22 to 3.30, suggesting that these algorithms are potentially capable of event-classification.

## **CHAPTER FOUR: Statistical Analysis**

### **4.1 Introduction**

Statistical analysis is applied in many fields such as science, math, economics, and humanities. Fundamentally, this type of analysis is applied to a collection of data, referred to as a “dataset”, to extract pertinent quantifiable characteristics. Statistical analysis is often performed to assist in making informed, rational decisions in many fields (e.g. Walpole et al., 2002).

This chapter describes the application of statistical analysis to microseismic signal classification. The developed algorithms focus on simplicity, functionality, and computational efficiency.

Compared to noise traces, I observed that “good” microseismic traces generally have lower signal variance, higher central data distribution, less frequent oscillations, and smaller signed amplitude differences between adjacent time-series data points. Statistical analysis classification algorithms were developed based on these observations. These algorithms are practical and simplified applications of the mathematics outlined in section 4.2 that describe fundamental statistical relationships.

### **4.2 Statistical Characteristics of Microseismic Signals**

Observed statistical characteristics of “good” and noise microseismic signals will first be described. Fundamental statistics theory will be discussed. Based on these characteristics, algorithms were developed to differentiate between “good” and noise microseismic signals through analysis of individual data points contained in the signals’ time series.

Figure 4.1 is an example “good” signal shown in Figure 1.7 (Event #1), reprinted for easier reference in this chapter. Figure 4.2 is an example noise signal shown in Figure 1.17 (Event #11).

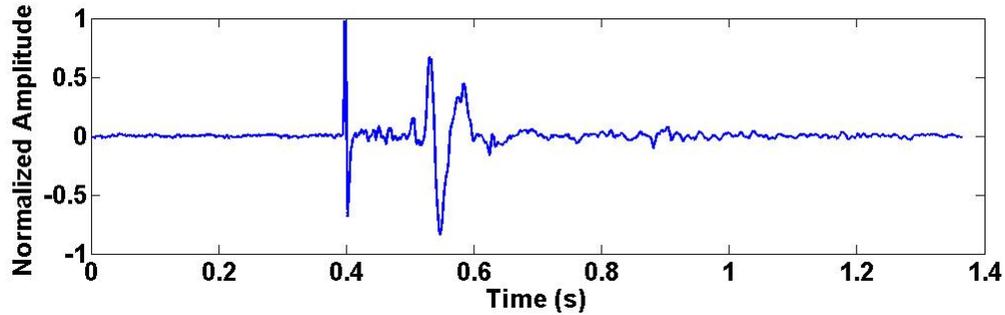


Figure 4.1: Example “good” signal.

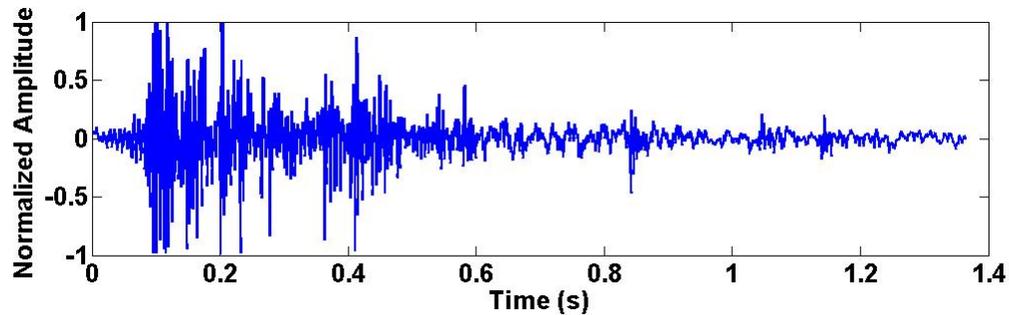


Figure 4.2: Example noise signal.

#### 4.2.1 Signal Variance

I observed that, in the Cold Lake dataset, microseismic noise signals tend to have greater data variance than “good” signals. As an example, define  $g_i(n_i)$  as the signed amplitude of the  $i^{\text{th}}$  time series data point corresponding to the “good” (noise) trace shown in Figure 4.1 (4.2), for  $i = 1, 2, \dots, N$ , where  $N$  is the total number of data points in the time series. Referring to Figures 4.1 and 4.2, a value of  $i = 1$  corresponds to a time of  $t_1 = \frac{1}{f_s}$ ,

and a value of  $i = N$  corresponds to a time of  $t_N = \frac{N}{f_s}$ , where  $f_s$  is the sampling frequency of the traces in Hertz. In Figures 4.1 and 4.2,  $N = 4096$ , and  $f_s = 3000$  Hz. Define  $\mu_g$  ( $\mu_n$ ) as the average of the example “good” (noise) time series. As an example,  $\mu_g$  can be expressed as

$$\mu_g = \frac{\sum_{i=1}^N g_i}{N}. \quad (4.1)$$

The mean of the noise trace  $\mu_n$  simply corresponds to replacing  $g$  with  $n$  in equation 4.1.

Define variance of the time series corresponding to the “good” (noise) trace as  $\sigma_g^2$  ( $\sigma_n^2$ ). Then,

$$\sigma_g^2 = \frac{\sum_{i=1}^N (g_i - \mu_g)^2}{N - 1}. \quad (4.2)$$

The variance of the noise trace  $\sigma_n^2$  simply corresponds to replacing  $g$  with  $n$  in equation 4.2.

The traces shown in Figures 4.1 and 4.2 have had DC offset removed, signifying that they have been adjusted such that  $\mu_g = \mu_n = 0$ . Thus, equation 4.2 can be reduced to

$$\sigma_g^2 = \frac{\sum_{i=1}^N g_i^2}{N - 1}. \quad (4.3)$$

Variance is a measure of statistical dispersion in a dataset (e.g. Walpole et al., 2002). It is a quantitative indicator of how values vary about the mean. Larger (smaller) dataset variances correspond to larger (smaller) expected deviations of arbitrary data points from the mean. The “good” microseismic trace in Figure 4.1 has a variance  $\sigma_g^2 = 0.0128$ ,

while the noise trace in Figure 4.2 has a variance  $\sigma_n^2 = 0.0293$ . By empirically examining these two figures, a higher noise trace variance would be expected. Over the whole trace, the noise trace contains visibly larger deviations from the mean compared to the “good” trace. Thus, one could intuitively conclude that more noise data points would be expected to lie outside an arbitrary mean-centered amplitude window than “good” data points. Section 4.2.1.1 below quantifies this intuitive conclusion to a greater degree.

#### 4.2.1.1 Chebyshev’s Inequality

The likelihood of dataset points existing a given distance away from its mean can be quantified by applying “Chebyshev’s inequality”. Theory pertaining to Chebyshev’s inequality is given in this section, adapted from Lange (2003); Therrien and Tummala (2004); Miller and Childers (2004); Mitzenmacher and Upfal (2005); and Suhov and Kelbert (2005), written in a compact and paraphrased form. Conclusions drawn will be applied to develop a microseismic signal classification algorithm.

Chebyshev’s inequality follows from Markov’s inequality, which is explained below.

Assume that  $\beta$  is a nonnegative arbitrary random variable ( $\beta \geq 0$ ). Define  $\beta_i$  as a single possible value of  $\beta$  with an occurrence probability of  $\rho_i$ . Then, the expected value of  $\beta$ ,  $\mathbf{E}[\beta]$ , can be defined as

$$\mathbf{E}[\beta] = \sum_i \rho_i \beta_i. \quad (4.4)$$

Define another variable  $\zeta$  that is related to  $\beta$  as

$$\begin{aligned} \xi &= 1 && \beta \geq a, \\ \xi &= 0 && \text{otherwise.} \end{aligned} \quad (4.5)$$

In equation 4.5,  $a$  is a positive arbitrary constant ( $a > 0$ ). Since  $\beta \geq 0$  and  $a > 0$ , the relationship

$$\xi \leq \frac{\beta}{a} \quad (4.6)$$

will hold. Since  $\xi$  is a variable that can only assume values of 0 or 1, referring to equation 4.5, the expected value of  $\xi$ ,  $\mathbf{E}[\xi]$ , will be

$$\mathbf{E}[\xi] = 0 \cdot \Pr(\xi = 0) + 1 \cdot \Pr(\xi = 1) = \Pr(\xi = 1) = \Pr(\beta \geq a), \quad (4.7)$$

where “ $\Pr(\xi = 1)$ ”, for example, means “the probability that  $\xi$  equals one”. Taking the expectation  $\mathbf{E}[\cdot]$  of both sides of equation 4.6 and applying the result of equation 4.7 gives

$$\Pr(\beta \geq a) = \mathbf{E}[\xi] \leq \mathbf{E}\left[\frac{\beta}{a}\right] = \frac{\mathbf{E}[\beta]}{a}, \quad (4.8)$$

where the well-known expectation relationship  $\mathbf{E}\left[\frac{\beta}{a}\right] = \frac{\mathbf{E}[\beta]}{a}$  was applied in equation 4.8.

The result

$$\Pr(\beta \geq a) \leq \frac{\mathbf{E}[\beta]}{a}, \quad (4.9)$$

is known as “Markov’s inequality”, and is a necessary relationship to obtain Chebyshev’s inequality.

Assume it is desired to determine the likelihood that a random variable  $X$  will lie, at the least, an absolute distance of  $a$  away from its mean  $\mathbf{E}[X]$ . Thus, it is desired to quantify  $\Pr(|X - \mathbf{E}[X]| \geq a)$ . Squaring both sides of this probability inequality leads to

$$\Pr(|X - \mathbf{E}[X]| \geq a) = \Pr((X - \mathbf{E}[X])^2 \geq a^2). \quad (4.10)$$

Observing that  $\{(X - \mathbf{E}[X])^2 \geq a^2\}$  allows the application of Markov's inequality defined in equation 4.9, which leads to

$$\Pr((X - \mathbf{E}[X])^2 \geq a^2) \leq \frac{\mathbf{E}[(X - \mathbf{E}[X])^2]}{a^2}. \quad (4.11)$$

Define  $\mathbf{VAR}[X]$  as the variance of variable  $X$ . The well-known identity

$$\mathbf{VAR}[X] = \mathbf{E}[(X - \mathbf{E}[X])^2] \quad (4.12)$$

will be applied. Combining equations 4.10, 4.11 and 4.12 leads to Chebyshev's inequality:

$$\Pr(|X - \mathbf{E}[X]| \geq a) \leq \frac{\mathbf{VAR}[X]}{a^2}. \quad (4.13)$$

Equation 4.13 states that a larger dataset variance corresponds to an increase in the expected maximum number of data points lying outside a mean-centered window of width  $2a$  (or single-sided width  $a$ ).

Recall from section 4.2.1, that the observed data variances in the “good” and noise trace examples were  $\sigma_g^2 = 0.0128$  and  $\sigma_n^2 = 0.0293$ , respectively. Equation 4.13 mathematically confirms and quantifies the intuitive conclusion that more noise data points would be expected to lie outside an arbitrary mean-centered amplitude window than “good” data points, since  $\sigma_n^2 > \sigma_g^2$ . The developed “Threshold” classification algorithm described in section 4.3.1 draws on this conclusion.

### 4.2.2. Signal Distribution

I observed that the time series signed amplitude distribution of “good” traces tended to be more heavily concentrated near the time axis than noise traces, as can be seen in the “good” and noise traces in Figures 4.1 and 4.2, respectively. One tool that can be used to examine and quantify data distribution is a histogram.

A histogram is a graphical representation of the number of signal data points that fall within disjointed amplitude ranges. To better illustrate this, assume that the signed amplitude of the  $i^{th}$  data point in an  $N$ -point discrete time series is  $d_i$ . For example, this time series could be one of the microseismic signals shown in Figures 4.1 or 4.2. Assume that both positive and negative values are present in this time series. Let  $d_{max}$  denote the largest positive value in the time series, and  $d_{min}$  denote the largest negative value. Assume there are  $n$  histogram bins each with a bin width of  $b$ . In this simple case, the total number of histogram bins,  $n$ , can then be calculated as

$$n = \left\lceil \frac{d_{max} - d_{min}}{b} \right\rceil, \quad (4.14)$$

where “ $\lceil \quad \rceil$ ” signifies rounding upwards to the closest integer. Let  $m_k$  represent histogram bin  $k$  for  $k = 0, 1, 2, \dots, n-1$ . One possible method is to position the lower boundary of the first bin  $m_0$  on  $d_{min}$ , the second bin  $m_1$  on  $d_{min} + b$ , and so on. Thus,  $m_k$  would record the total number of points that fall in the range  $d_{min} + kb \leq d_i < d_{min} + (k+1)b$ . The notation

$$m_k \rightarrow [d_{min} + kb \leq d_i < d_{min} + (k+1)b]; \quad k = 0, 1, 2, \dots, n-1 \quad (4.15)$$

will be used to signify this.

An issue that arises when creating a histogram includes determining the optimal bin width  $b$  (or equivalently, through equation 4.14, the total number of bins  $n$ ). It is desired to

choose  $b$  (or  $n$ ) such that the data distribution is well represented. If  $b$  is too large, critical information may be smeared across a single bin. That is, the frequency structure of the data may not be appropriately emphasized. Conversely, if  $b$  is too small, fluctuations in the histogram that may appear important could be the simple result of random data variation over a small interval.

Statistics literature has attempted to determine how to optimally choose  $b$ . Three well-known equations have been developed by Sturges (1926), Scott (1979), as well as Freedman and Diaconis (1981). These equations will be briefly discussed.

Sturges (1926) assumed a histogram containing a total of  $n$  bins, where the number of data points recorded in histogram bin  $k$  ( $k = 0, 1, 2, \dots, n-1$ ) was equal to the “binomial coefficient”  $\binom{n-1}{k}$ . Using the general equation

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (4.16)$$

(e.g. Walpole et al., 2002), where “!” is the factorial operator, the number of data points in histogram bin  $k$  can be rewritten as

$$\binom{n-1}{k} = \frac{(n-1)!}{k!(n-1-k)!} \quad (4.17)$$

Thus, the total number of points in the entire dataset (sample size),  $N$ , is the summation of the number of points contained in each bin, expressed as

$$N = \sum_{k=0}^{n-1} \binom{n-1}{k} \quad (4.18)$$

Sturges (1926) chose to assume this type of data distribution because as

$n$  increases to large values, the data distribution approaches the shape of a normal (or Gaussian) distribution, which can be a reasonable approximation for some datasets. A normal distribution of a random variable  $X$ , that can assume values of  $x$  ( $x \in \mathfrak{R}$ ), is one whose probability density function  $f(x)$  is given as

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad (4.19)$$

where  $\mu$  is the dataset mean, and  $\sigma^2$  is the dataset variance. As the number of bins  $n$  increases, the assumed data distribution approaches the shape of the Gaussian distribution in equation 4.19 with  $\mu = \frac{n-1}{2}$  and  $\sigma^2 = \frac{n-1}{4}$  (e.g. Scott, 1992).

The “binomial theorem” is a well-known equation that expands out powers of sums. In its simplest and general form, it is given as

$$(x+y)^n = \sum_{k=0}^n \binom{n}{k} x^{n-k} y^k \quad (4.20)$$

(e.g. Walpole et al., 2002). Equation 4.18 can be rewritten by comparing it to equation 4.20 with  $x = y = 1$ . This results in the sample size expressed as

$$N = 2^{n-1}. \quad (4.21)$$

Taking the logarithm of equation 4.21, the theoretical optimum number of bins  $n$  corresponding to the assumption of Sturges (1926) is then

$$n = \log_2 N + 1. \quad (4.22)$$

Equation 4.22 is known as “Sturges’ rule”, which provides a suggested number of histogram bins  $n$  given a dataset with sample size  $N$ .

While Sturges' rule is often used as a rule of thumb for quick histogram generation, it is not universally applicable to all datasets (e.g. Scott, 1992). Sturges' rule implicitly assumes a normal data distribution (for large  $n$ ), an approximation that is appropriate for some datasets, but not for others. It has also been shown that as the sample size  $N$  becomes large, Sturges' rule tends to result in histograms that are overly smoothed, resulting in information smeared across single bins.

Alternative equations to determine the histogram bin width  $b$  have been developed in statistics literature to improve upon Sturges' rule. Scott (1979) developed the relationship

$$b = \frac{3.5s}{\sqrt[3]{N}}. \quad (4.23)$$

Freedman and Diaconis (1981) developed the expression

$$b = \frac{2Q_{13}}{\sqrt[3]{N}}. \quad (4.24)$$

In equation 4.23,  $s$  corresponds to the sample standard deviation. In equation 4.24,  $Q_{13}$  pertains to the sample interquartile range, which is the difference between the two data sample values on the inner limits of the first and third quartiles of the dataset.

While equations 4.23 and 4.24 improve upon Sturges' rule, some limitations still exist. For instance, He and Meeden (1997) demonstrate an example where Scott's relationship (equation 4.23) has the potential of obscuring histogram structure when the data is roughly distributed (as opposed to smoothly distributed).

Generally, if time permits, it is best to empirically select histogram bin widths catered to the specific dataset under examination. No mathematical relationship concerning optimal histogram bin width has been developed that is optimally applicable to all types of

datasets. Thus, the bin width corresponding to the “Histogram” microseismic file classification algorithm described in section 4.3.2 was determined empirically.

#### 4.2.3 Signal Oscillation and Sequential Time-Series Behavior

I observed that microseismic noise signals tend to oscillate more frequently about the time axis. This is similar to the frequency observations discussed in chapter 2. However, it was additionally observed that the magnitudes of signed amplitude differences between adjacent time-series data points were generally greater for noise microseismic traces compared to “good” traces. Essentially, this observation implies that “good” traces generally contain minimal signal noise. Noise that was observed in microseismic “good” traces had relatively low amplitude compared to microseismic noise traces. Alternatively stated, it was observed that, compared to noise traces, “good” traces often have less sporadic sequential time-series behaviour about its mean.

To illustrate the above observations, the “good” and noise traces in Figures 4.1 and 4.2 are plotted as discrete time-series in Figures 4.3 and 4.4 from an example time interval of 1.14s to 1.15s. A very fine time interval is used in these figures to illustrate the signed amplitude difference between adjacent time-series data points corresponding to the example “good” and noise traces. Note that in Figure 4.4, corresponding to the noise trace, there are more instances where adjacent data points have opposite signs compared to the “good” trace in Figure 4.3. This is related to the observation that “good” traces oscillate less than noise traces. An additional important observation is that the magnitudes of signed amplitude differences between adjacent time-series data points are larger for the noise trace than the “good” trace, including areas where adjacent data points change polarity. Based on

these observations the “Specialized Zero-Crossing Count” algorithm described in section 4.3.3 was developed.

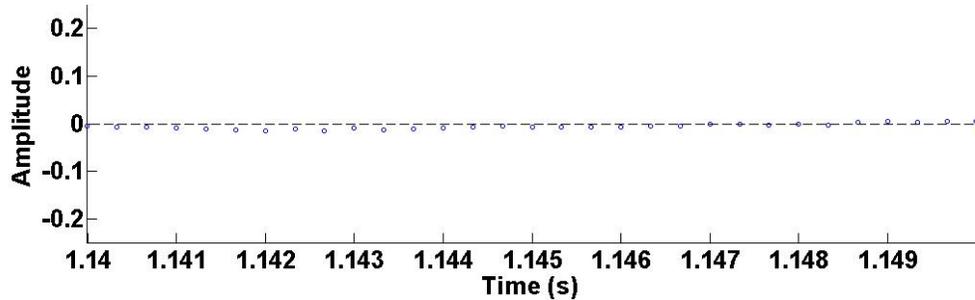


Figure 4.3: Discrete time-series plot of example “good” trace for  $t = 1.14\text{s}$  to  $t = 1.15\text{s}$ . The dashed line represents zero amplitude to better view polarity reversals. Fewer polarity reversals between adjacent data points are seen compared to the noise trace in Figure 4.4. Additionally, the magnitudes of signed amplitude differences between adjacent time-series data points are smaller compared to the example noise trace.

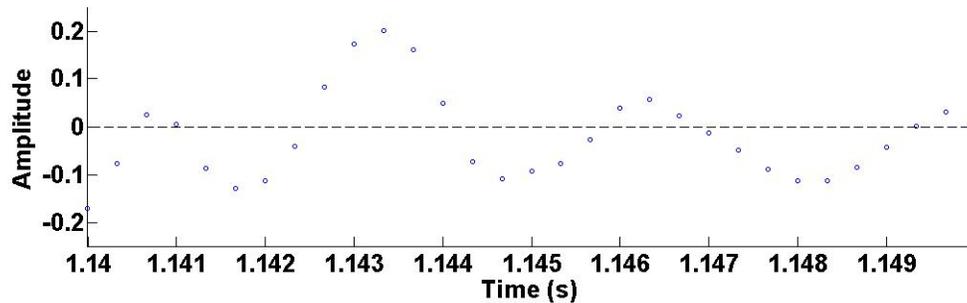


Figure 4.4: Discrete time-series plot of example noise trace for  $t = 1.14\text{s}$  to  $t = 1.15\text{s}$ . The dashed line represents zero amplitude to better view polarity reversals. More polarity reversals between adjacent data points are seen compared to the “good” trace in Figure 4.3. Additionally, the magnitudes of signed amplitude differences between adjacent time-series data points are larger compared to the example “good” trace.

### 4.3 Statistical Classification Algorithms

Based on the observed signal characteristics described in sections 4.2.1 to 4.2.3, simple statistical algorithms were developed to classify microseismic files. Statistical analysis algorithms that were both simple and functional were developed to reduce computational time. In practical applications, minimal computation time is critical, as potentially tens of thousands of microseismic files may need to be classified daily. The developed statistical analysis algorithms are practical and simplified applications of the statistical mathematics outlined in section 4.2. It will be shown in chapter 5 that these statistical analysis techniques have the potential to classify microseismic files to a high degree of accuracy.

#### 4.3.1 Threshold Algorithm

As described in section 4.2.1, it was concluded that more noise data points would be expected to lie outside an arbitrary mean-centered amplitude window than “good” data points, since  $\sigma_n^2 > \sigma_g^2$ . This “Threshold” algorithm draws on this conclusion.

Define a window  $w$  existing from  $-a \leq w \leq a$ . This window is centered at the mean of examined traces, which is zero. This algorithm simply determines the fraction of time series data points that lie outside the threshold limits  $\pm a$  of the window  $w$ . This fraction can then be used for microseismic signal classification. Table 4.1 summarizes the percentage of outlying data points using a single-sided window width of  $a = 0.03$  when this algorithm is applied to the synthetic traces shown in Figures 1.22 to 1.28. Table 4.2 summarizes the percentage of outlying data points with  $a = 0.03$  when this algorithm is applied to Event #1

through Event #15. Note that, compared to noise traces, the example “good” traces generally contain fewer outlying data points.

Table 4.1: Summary of the percentage of outlying data points using a single-sided window width of  $a = 0.03$  when Threshold algorithm is applied to the synthetic traces.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Pts Outside (%)</b>
Good	30	4.9
Good	25	22.7
Good	20	49.8
Good	15	59.3
Good	10	78.4
Good	0	89.2
Noise	N/A	70.3

Table 4.2: Summary of the percentage of outlying data points using a single-sided window width of  $a = 0.03$  when Threshold algorithm is applied to example “good” and noise events. Note that, compared to noise traces, the example “good” traces generally contain fewer outlying data points.

<b>Event Type</b>	<b>Event #</b>	<b>Pts Outside (%)</b>
Good	1	20.8
Good	2	40.6
Good	3	21.0
Good	4	39.2
Good	5	6.4
Good	6	42.4
Good	7	52.2
Good	8	34.2
Good	9	54.5
Good	10	13.2
Noise	11	68.2
Noise	12	73.2
Noise	13	79.3
Noise	14	31.7
Noise	15	65.8

### 4.3.2 Histogram Algorithm

Referring to the histogram theory presented in section 4.2.2, and the observation that the time series signed amplitude distribution of “good” traces tended to be more heavily concentrated near the time axis than noise traces, a “Histogram” microseismic classification technique was developed.

The histogram parameters were empirically chosen to best fit the given microseismic dataset. A total of  $n = 99$  histogram bins, each of width  $b = \frac{2}{99}$  were applied. The data limits were taken to be  $d_{min} = -1$  and  $d_{max} = +1$  to obtain a symmetrical histogram about the zero-mean signals. Referring to equation 4.15, histogram bin  $k$  ( $k = 0, 1, 2, \dots, 98$ ), represented by  $m_k$ , would thus record the total number points that fall in the range

$-1 + \frac{2k}{99} \leq d_i < -1 + \frac{2(k+1)}{99}$ . Using the notation in equation 4.15 for this example gives

$$m_k \rightarrow \left[ -1 + \frac{2k}{99} \leq d_i < -1 + \frac{2(k+1)}{99} \right]; \quad k = 0, 1, 2, \dots, 98. \quad (4.25)$$

After histogram generation, the data occurrence recorded in a central bin (corresponding to  $k \approx \frac{n}{2}$ , roughly speaking) can be examined for microseismic signal classification.

If the 50<sup>th</sup> bin ( $k = 49$ ) is examined, the synthetic “good” traces in Figures 1.22 to 1.27, which had SNR decreasing from 30 dB to 0 dB, had 32.4%, 18.1%, 10.4%, 5.93%, 4.03%, and 2.37% of data points lying in the 50<sup>th</sup> bin range, respectively. This is expected because data point distribution spreads out to larger degrees and becomes less centralized as noise power increases. The synthetic noise event shown in Figure 1.28 has 1.1% of its data points in the 50<sup>th</sup> bin range. These results are summarized in Table 4.3. Table 4.4

summarizes the percentage of data points that lie in the 50<sup>th</sup> bin range pertaining to Event #1 through Event #15 when this algorithm is applied. Note that, compared to “good” events, noise events generally have fewer data points that lie in the 50<sup>th</sup> bin range. Figures 4.5 to 4.26 show histogram plots corresponding to the “good” and noise synthetic and real traces.

Table 4.3: Summary of the percentage of data points in the 50<sup>th</sup> (middle) bin range when the Histogram algorithm is applied to the synthetic traces.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Pts in 50<sup>th</sup> bin (%)</b>
Good	30	32.4
Good	25	18.1
Good	20	10.4
Good	15	5.93
Good	10	4.03
Good	0	2.37
Noise	N/A	1.1

Table 4.4: Summary of the percentage of data points in the 50<sup>th</sup> (middle) bin range when the Histogram algorithm is applied to example “good” and noise events. Note that, compared to “good” events, noise events generally have fewer data points that lie in the 50<sup>th</sup> bin range.

<b>Event Type</b>	<b>Event #</b>	<b>Pts in 50<sup>th</sup> bin (%)</b>
Good	1	34.6
Good	2	29.7
Good	3	33.6
Good	4	22.8
Good	5	48.5
Good	6	18.9
Good	7	12.0
Good	8	25.9
Good	9	15.1
Good	10	43.5
Noise	11	10.7
Noise	12	8.9
Noise	13	6.6
Noise	14	33.8
Noise	15	10.4

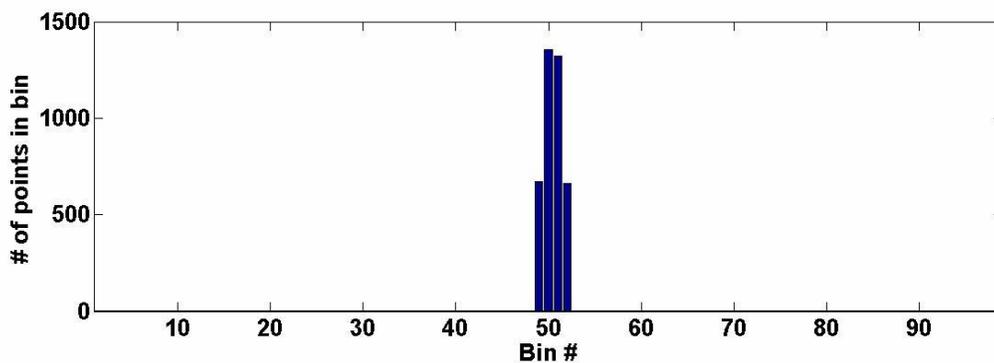


Figure 4.5: Histogram plot corresponding to synthetic “good” trace (SNR = 30 dB) shown in Figure 1.22.

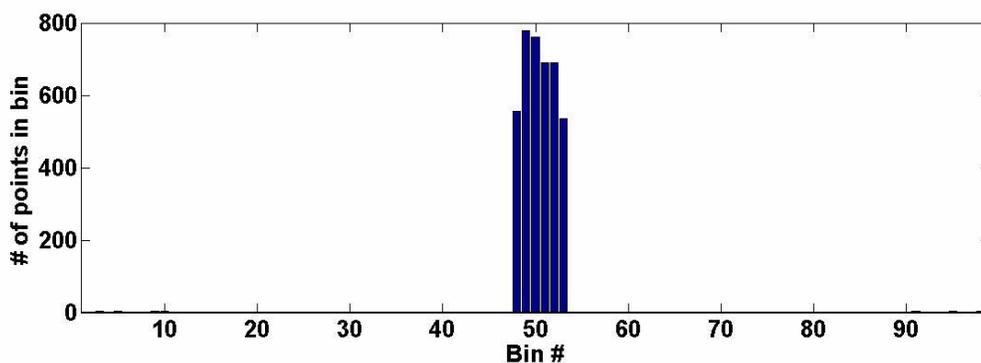


Figure 4.6: Histogram plot corresponding to synthetic “good” trace (SNR = 25 dB) shown in Figure 1.23.

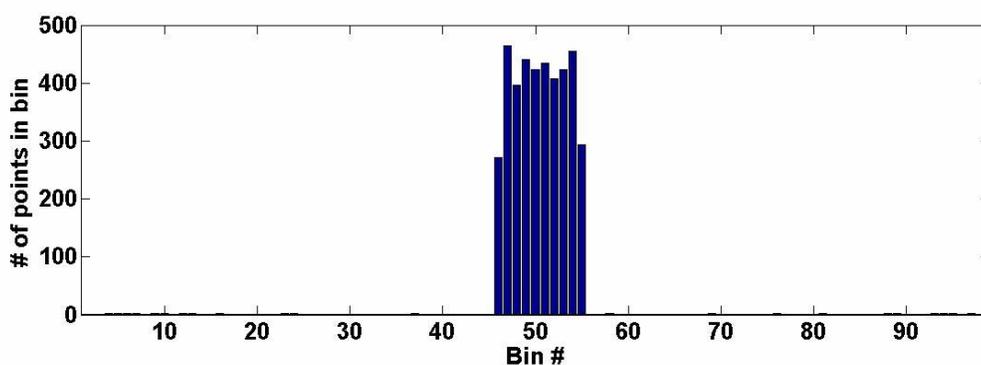


Figure 4.7: Histogram plot corresponding to synthetic “good” trace (SNR = 20 dB) shown in Figure 1.24.

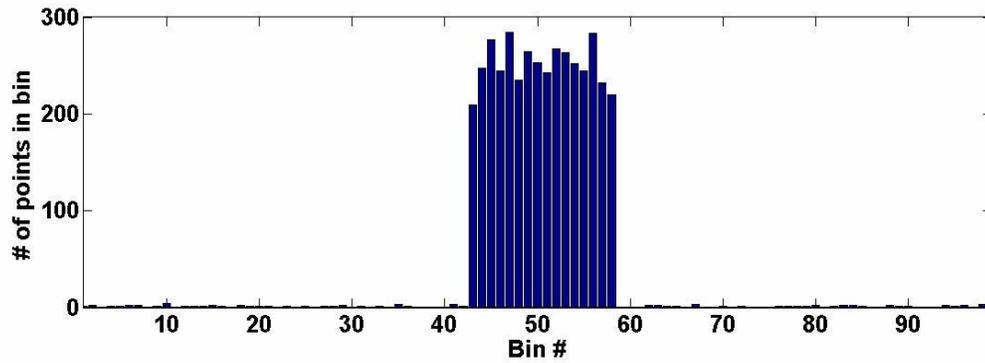


Figure 4.8: Histogram plot corresponding to synthetic “good” trace (SNR = 15 dB) shown in Figure 1.25.

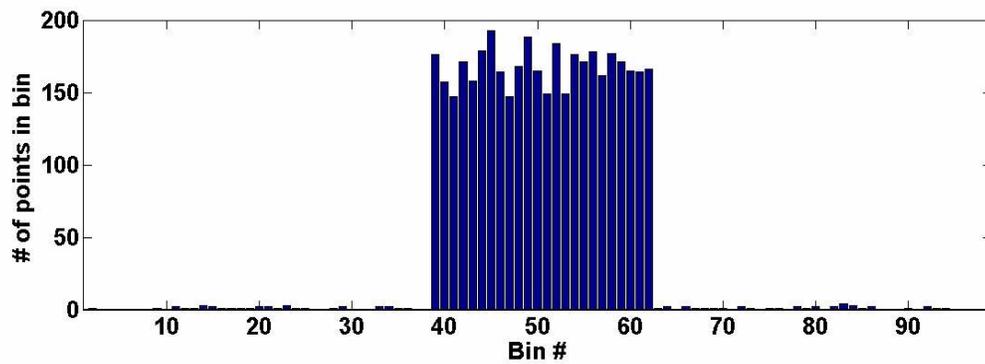


Figure 4.9: Histogram plot corresponding to synthetic “good” trace (SNR = 10 dB) shown in Figure 1.26.

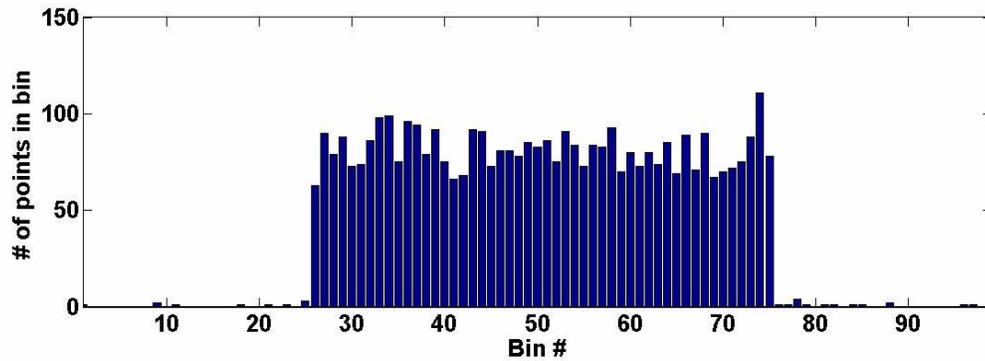


Figure 4.10: Histogram plot corresponding to synthetic “good” trace (SNR = 0 dB) shown in Figure 1.27.

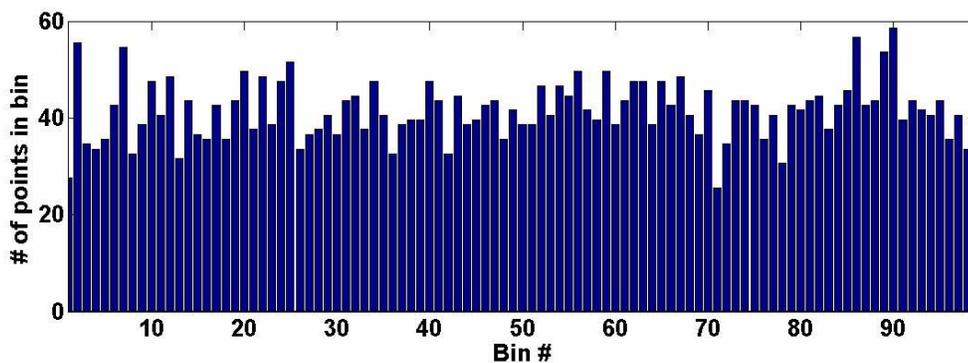


Figure 4.11: Histogram plot corresponding to synthetic noise event shown in Figure 1.28.

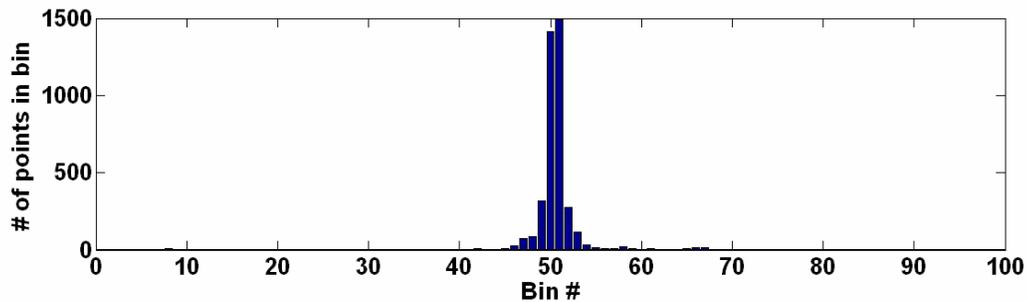


Figure 4.12: Histogram plot corresponding to Event #1, a “good” trace.

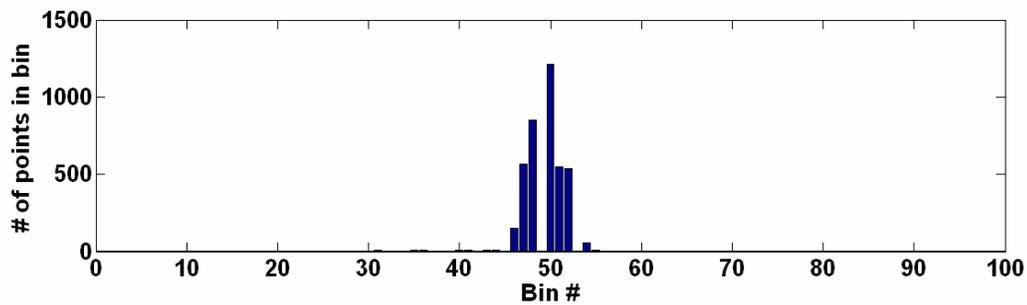


Figure 4.13: Histogram plot corresponding to Event #2, a “good” trace.

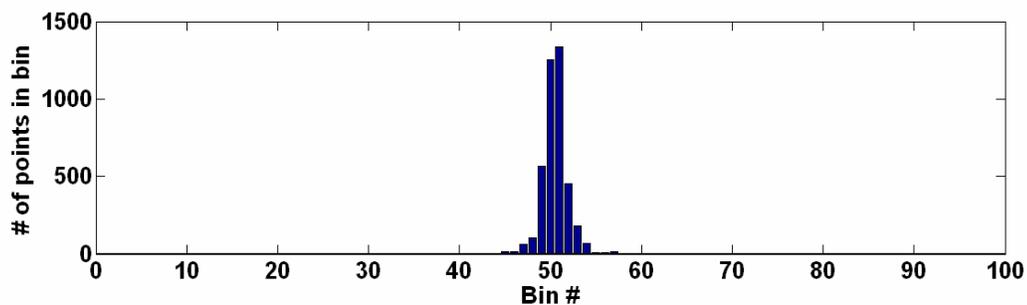


Figure 4.14: Histogram plot corresponding to Event #3, a “good” trace.

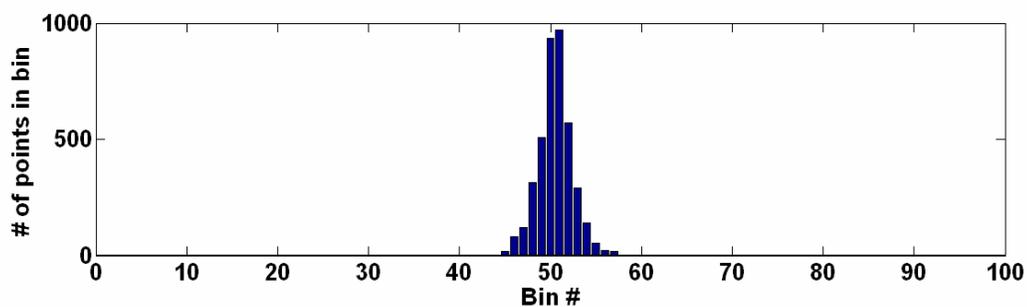


Figure 4.15: Histogram plot corresponding to Event #4, a “good” trace.

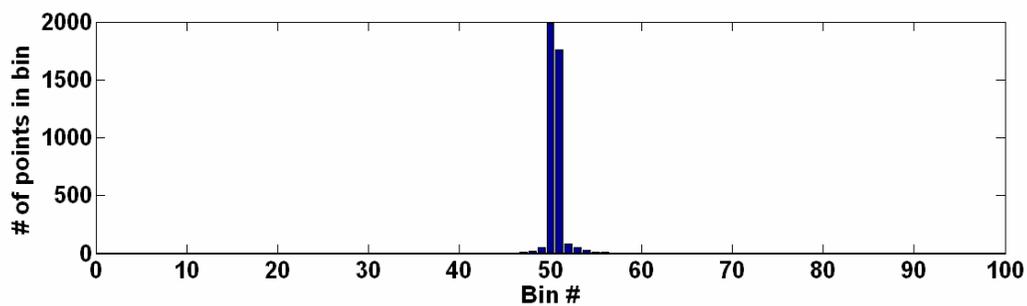


Figure 4.16: Histogram plot corresponding to Event #5, a “good” trace.

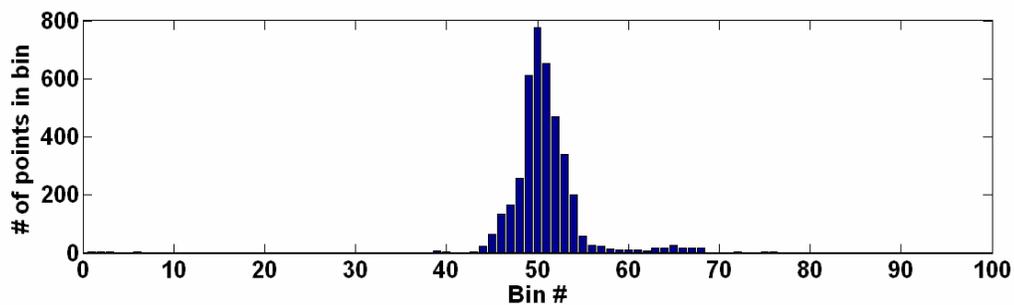


Figure 4.17: Histogram plot corresponding to Event #6, a “good” trace.

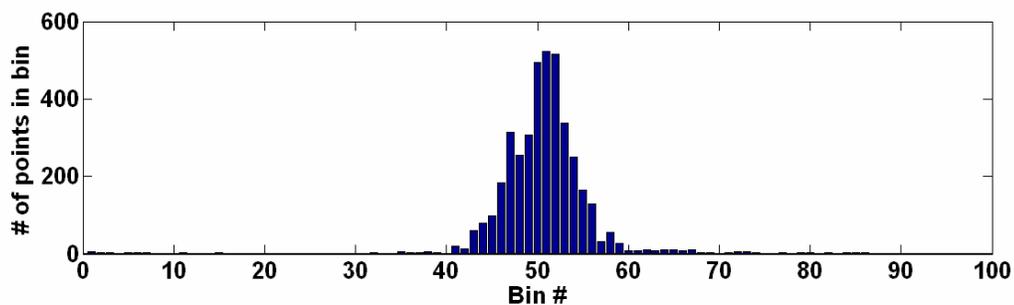


Figure 4.18: Histogram plot corresponding to Event #7, a “good” trace.

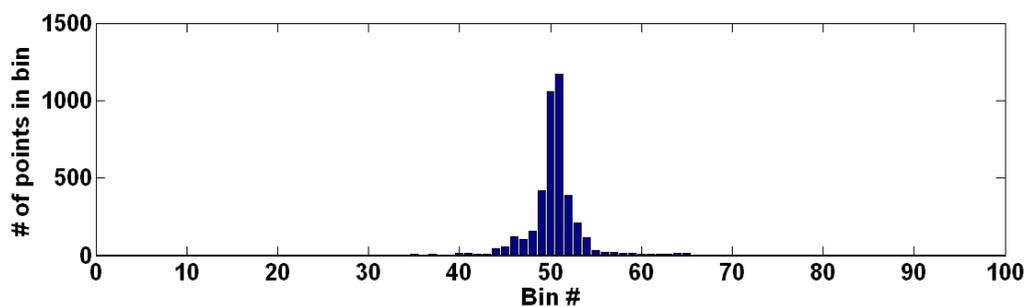


Figure 4.19: Histogram plot corresponding to Event #8, a “good” trace.

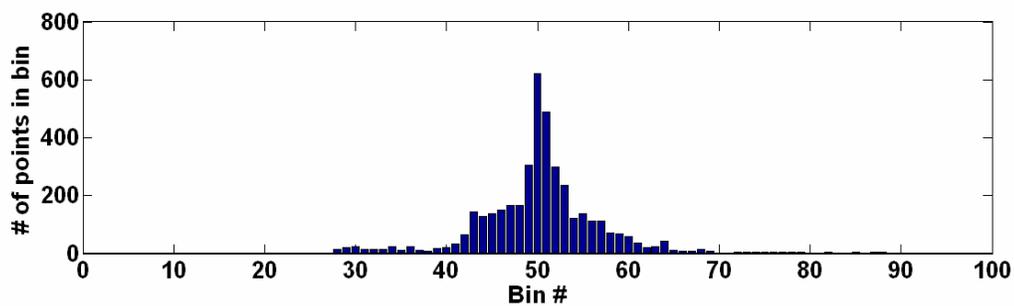


Figure 4.20: Histogram plot corresponding to Event #9, a “good” trace.

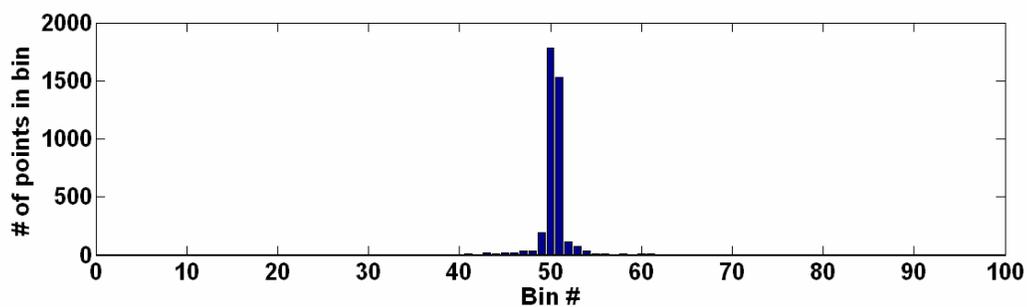


Figure 4.21: Histogram plot corresponding to Event #10, a “good” trace.

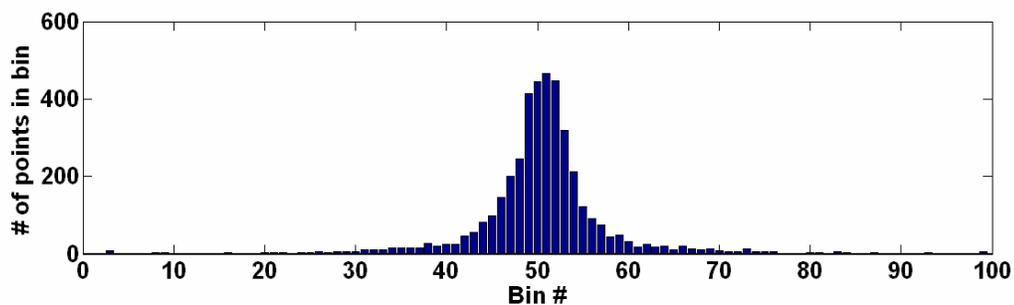


Figure 4.22: Histogram plot corresponding to Event #11, a noise trace.

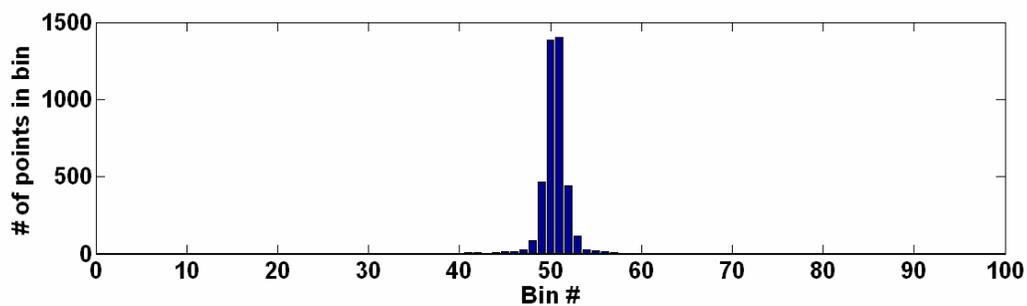


Figure 4.23: Histogram plot corresponding to Event #12, a noise trace.

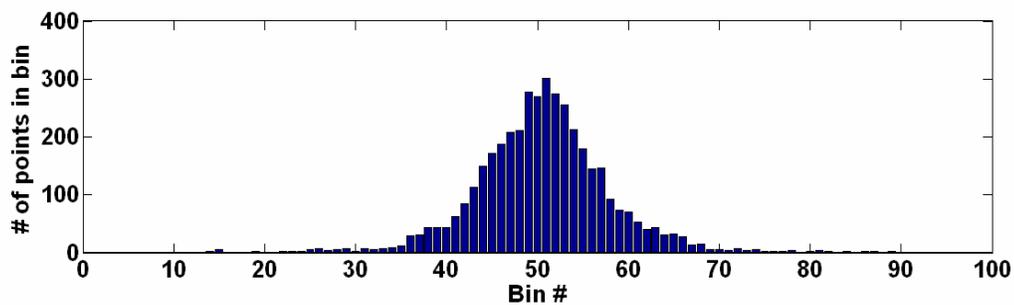


Figure 4.24: Histogram plot corresponding to Event #13, a noise trace.

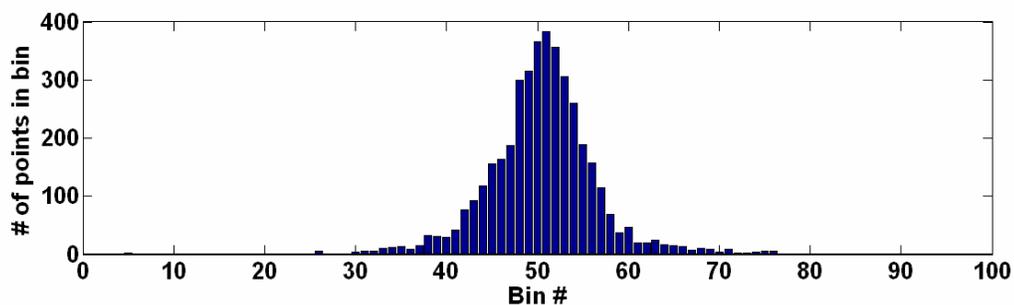


Figure 4.25: Histogram plot corresponding to Event #14, a noise trace.

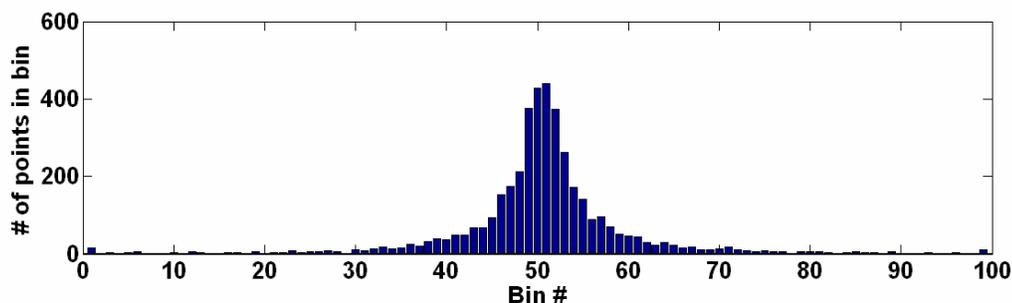


Figure 4.26: Histogram plot corresponding to Event #15, a noise trace.

### 4.3.3 Specialized Zero-Crossing Count Algorithm

Based on the observations that microseismic noise signals tend to oscillate more frequently about the time axis and that magnitudes of signed amplitude differences between adjacent time-series data points were generally greater compared to “good” traces, a “Specialized Zero-Crossing Count” algorithm was developed.

Define a window  $v$  that exists in the signed-amplitude range  $-z \leq v \leq z$ . First, all the time-series data points that fall within  $v$  are set to zero. Essentially, this zeroing step sets all data points close to the time axis to zero and removes low-amplitude noise. As the signed amplitudes of adjacent time-series data points in microseismic noise signals generally vary to greater degrees compared to “good” traces, this zeroing step will tend to preserve polarity reversals (sign changes) in adjacent data points on noise traces, but eliminate many of these polarity reversals on “good” traces. Thus, this step helps to further improve the discrepancy between “good” and noise traces to improve classification accuracy.

After the zeroing step is applied, the total number of valid polarity reversals between adjacent data points is totaled, and that total is divided by the total number of trace data points to obtain a fractional measurement. A valid polarity reversal corresponds to

adjacent data points changing from a strictly positive to a strictly negative value, or vice-versa.

With the zeroing step applied for  $z = 0.01$ , the synthetic “good” traces in Figures 1.22 to 1.27, which had SNR decreasing from 30 dB to 0 dB, had measured polarity reversal percentages of 15.7%, 26.9%, 31.3%, 36.3%, 38.4%, and 43.6%, respectively. The synthetic noise trace in Figure 1.28 had a measured polarity reversal percentage of 40.8%. These results are summarized in Table 4.5. Table 4.6 summarizes the results when this algorithm is applied to Event #1 through Event #15. Note that the “good” events generally have significantly fewer valid polarity reversals than noise events.

Table 4.5: Summary of the percentage of valid polarity reversals (zero-crossings) when the Specialized Zero-Crossing Count algorithm is applied to the synthetic traces.

<b>Event Type</b>	<b>SNR (dB)</b>	<b>Zero-Crossings (%)</b>
Good	30	15.7
Good	25	26.9
Good	20	31.3
Good	15	36.3
Good	10	38.4
Good	0	43.6
Noise	-	40.8

Table 4.6: Summary of the percentage of valid polarity reversals (zero-crossings) when the Specialized Zero-Crossing Count algorithm is applied to example “good” and noise events. Note that “good” events generally have significantly less valid polarity reversals than noise events.

<b>Event Type</b>	<b>Event #</b>	<b>Zero-Crossings (%)</b>
Good	1	0.0244
Good	2	3.42
Good	3	0.0977
Good	4	0.22
Good	5	0.147
Good	6	0.171
Good	7	0.0244
Good	8	0.195
Good	9	0.0732
Good	10	0.513
Noise	11	7.28
Noise	12	5.82
Noise	13	4.52
Noise	14	3.91
Noise	15	9.20

To illustrate the value of applying the zeroing step to eliminate low-amplitude noise present in the time series, consider an example case where this zeroing step is omitted. Event #1, a “good” trace, will be compared to Event #11, a noise trace, as an example. If the zeroing step is omitted, Event #1 would have 302 polarity reversals out of 4096 points (7.37%), while Event #11 would have 550 polarity reversals (13.43%). Thus, omitting the zeroing step, only 1.82 times more polarity reversals would be seen in the example noise trace (Event #11) compared to the “good” trace (Event #1). Referring to Table 4.6, Event #11 has 298 times more polarity reversals than Event #1 with the zeroing step applied.

#### 4.4 Conclusion

When compared to noise traces, I observed that many “good” microseismic traces in the Cold Lake dataset generally have lower signal variance, higher central data distribution, less frequent oscillations, and smaller signed amplitude differences between adjacent time-series data points. Based on these observations, statistical analysis algorithms were developed after fundamental statistical mathematics was reviewed.

The developed statistical algorithms are the “Threshold”, “Histogram”, and “Specialized Zero-Crossing Count” techniques described in this chapter. These techniques focus on functionality and simplicity for accurate and computationally efficient microseismic file classification.

Significant differences in algorithm measurements were obtained when these techniques were tested on example “good” and noise traces. These results suggest that the developed statistical analysis techniques are capable of reasonably accurate microseismic file classification.

## CHAPTER FIVE: Multivariate Data Reduction

### 5.1 Introduction

Testing was performed to determine which algorithms perform best concerning microseismic file classification. An example of these tests is discussed in section 5.3. After extensive testing, it was found that statistical analysis algorithms discussed in chapter four yield results with the highest potential for consistently accurate microseismic file classification. This was seen through statistical algorithms giving the most consistent clustered results when tested on “good” and noise files. These algorithms were the Threshold, Histogram, and Specialized Zero-Crossing Count methods.

It is desired to examine the outputs from the developed algorithms for accurate microseismic file classification. As these algorithm outputs result in a multidimensional data space, multivariate data reduction (i.e. reducing the effective dimensionality of the data) would simplify file classification. The multivariate data reduction technique employed is referred to as principal components analysis.

Principal components analysis was performed first using outputs from all developed algorithms, shown in section 5.4.1. Following this, principal components analysis was performed using only the three statistical algorithm outputs, discussed in section 5.4.2. It is shown that optimal microseismic file classification accuracy is obtained when principal components analysis is restricted to the statistical algorithm outputs.

## 5.2 Principal Components Analysis

Given a dataset with many measured variables, redundancy could be present. This redundancy is related to the mathematical correlation between variables in a multivariate dataset. It is often desired to remove redundancy between these variables to more easily extract key information from the dataset.

Principal components analysis (PCA) is a linear technique that transforms a dataset with many variables to a new set of variables that are orthogonal and uncorrelated, called the “principal components” of the dataset (e.g. Jackson, 1991). PCA can also be thought of as representing an  $N$  dimensional dataset with  $N$  orthogonal basis vectors (principal components) such that data projected onto the first principal component have the highest variance and thus best characterize the dataset (e.g. Shlens, 2003).

PCA has been applied in many fields, including astrophysics, neuroscience, criminology, and computer science (i.e. image processing / pattern recognition). Here, PCA was used to optimize microseismic file classification accuracy through application to the outputs of the developed algorithms.

Mathematical theory pertaining to PCA is discussed in Appendix C. Following the test results described in section 5.3, an application example of PCA to classification algorithm outputs is shown in section 5.4.

### 5.3 Test Results

A diverse example test dataset of 540 microseismic files from 28 different production pads is used in this section. The correct file classification is known, as these files were previously manually classified as “good” or noise.

Algorithm outputs after normalization were obtained for each file in the dataset. Normalization indicates that each algorithm measurement was divided by the largest value pertaining to that measurement over all examined microseismic files. Normalization is necessary when applying principal components analysis, a multivariate data reduction technique (refer to Appendix C).

For the frequency-filtering techniques, the normalized algorithm outputs correspond to the normalized peak amplitude of the trace after filtering described in chapter 2. The normalized algorithm output of the time and frequency-domain event-length detection techniques are the normalized calculated trace event-length described in chapter 3. The normalized algorithm outputs of the Threshold, Histogram, and Specialized Zero-Crossing Count statistical analysis techniques pertain to the normalized fraction of outlying data points (Threshold), the normalized fraction of data points falling in the range pertaining to a centre histogram bin (Histogram), and the normalized fraction of polarity reversals after low-amplitude noise is removed (Specialized Zero-Crossing Count) described in chapter 4. For the tests shown here, the algorithm settings pertaining to the statistical analysis algorithms are slightly modified from those given in chapter 4. The Threshold window was set to  $\pm 0.006$  (Threshold algorithm), the 51<sup>st</sup> histogram bin was examined (Histogram algorithm), and low-amplitude noise with an absolute value less than 0.006 was removed (Specialized Zero-Crossing Count algorithm).

Figures 5.1 to 5.8 show normalized algorithm outputs pertaining to the eight developed algorithms discussed in chapters 2 through 5. Each pink square (blue diamond) corresponds to an algorithm output pertaining to a known “good” (noise) file. The calculated and normalized results from the known “good” and noise files are plotted in overlapping horizontal regions to best view vertical overlap between the normalized “good” and noise results. Ideally, no overlap between calculated and normalized algorithm outputs of known “good” and noise files is desired to perform perfect classification. Each plotted data point represents the normalized average algorithm output corresponding to the three traces in the microseismic file that have the strongest “good” characteristics. For example, in Figure 5.6, which corresponds to the Threshold algorithm outputs, each point represents the normalized average fraction of data points lying outside a predefined window corresponding to the three traces in a microseismic file that contain the lowest fraction of outlying points.

From Figures 5.1 to 5.8, it can be seen that applying the statistical algorithms result in best data clustering, where there is minimal vertical overlap for algorithm measurements between “good” and noise files. Thus, it may be appropriate to restrict microseismic file classification by using statistical algorithm outputs only. Section 5.4 will illustrate principal components analysis first applied to all eight algorithm outputs, followed by an application of this technique to the three statistical algorithm outputs only. It is shown in section 5.4 that restricting principal components analysis to statistical algorithm outputs yields improved classification accuracy.

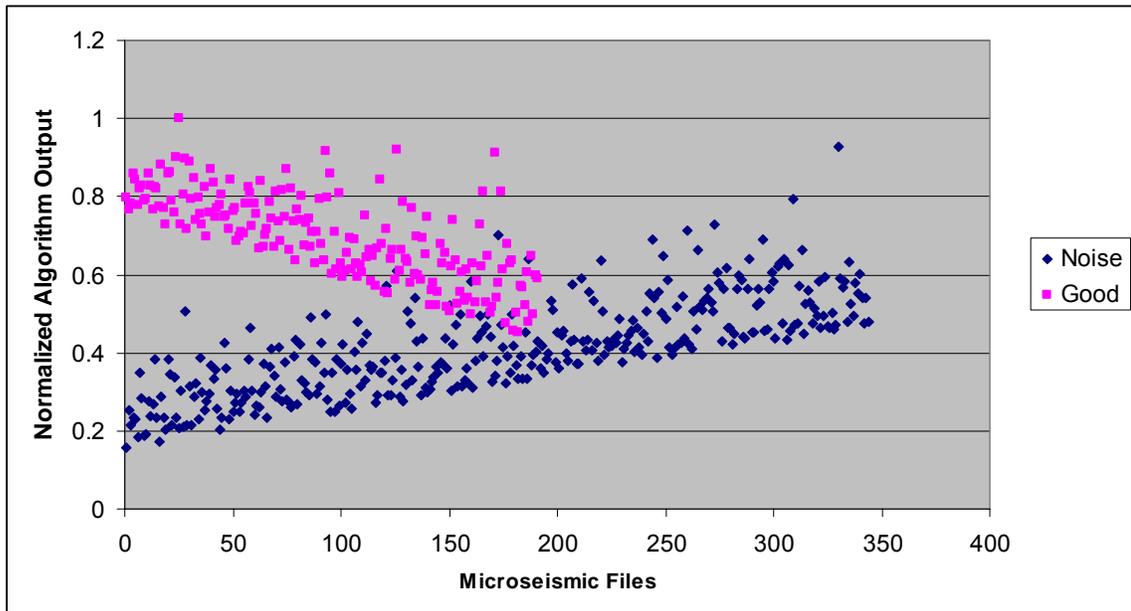


Figure 5.1: Normalized algorithm output of low-pass filtering algorithm described in section 2.3. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.

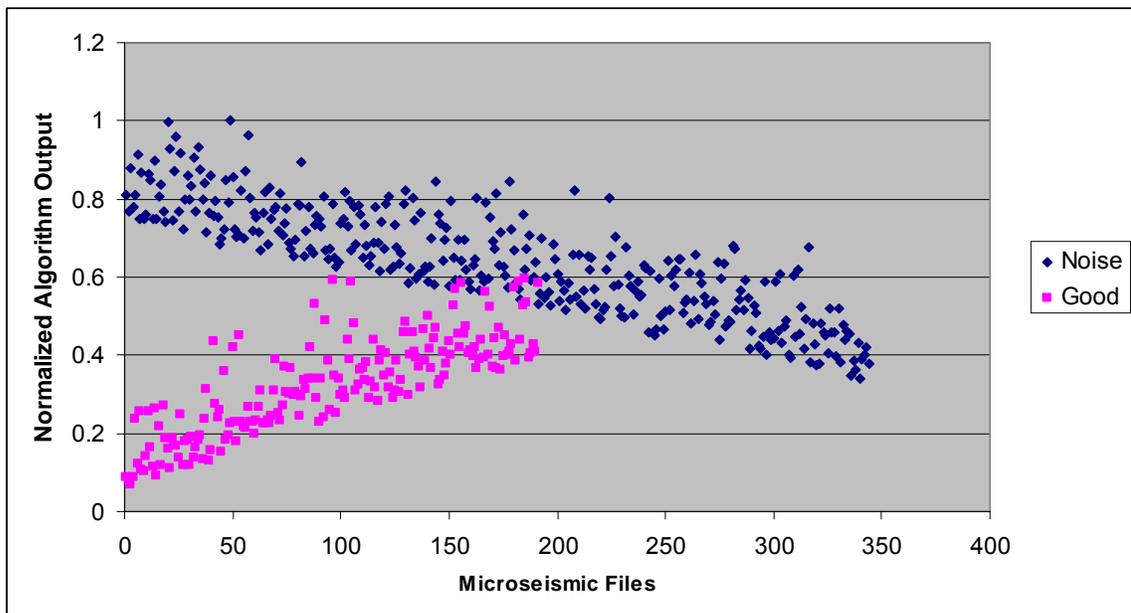


Figure 5.2: Normalized algorithm output of high-pass filtering algorithm described in section 2.4. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.

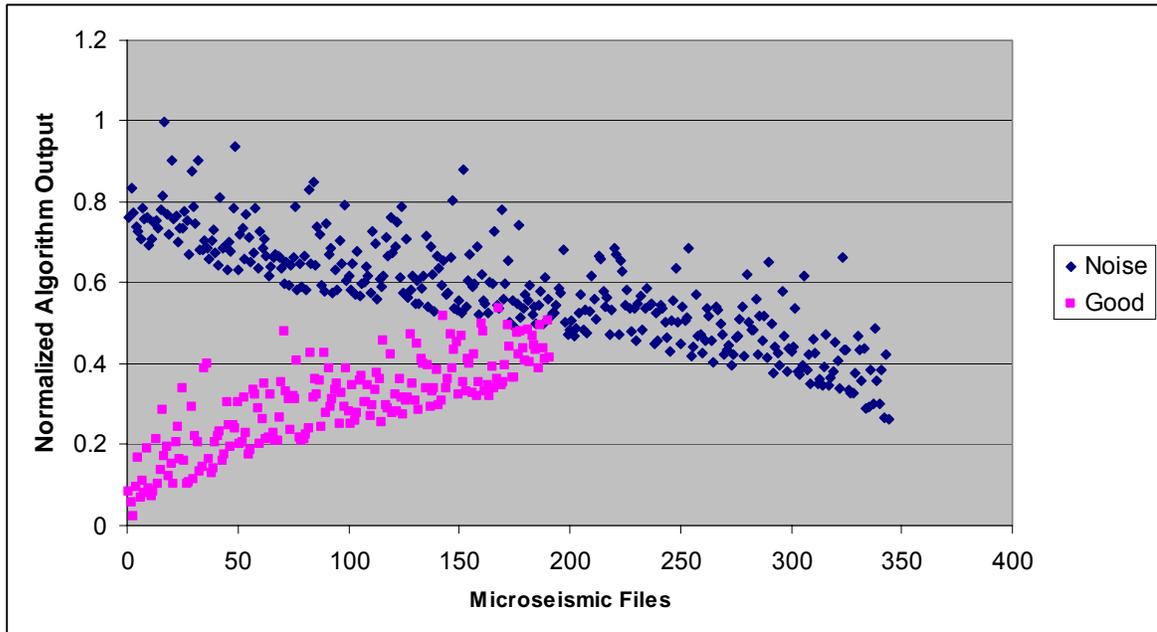


Figure 5.3: Normalized algorithm output of band-pass filtering algorithm described in section 2.5. Each pink square (blue diamond) pertains to a normalized peak amplitude measurement for a single “good” (noise) microseismic file.

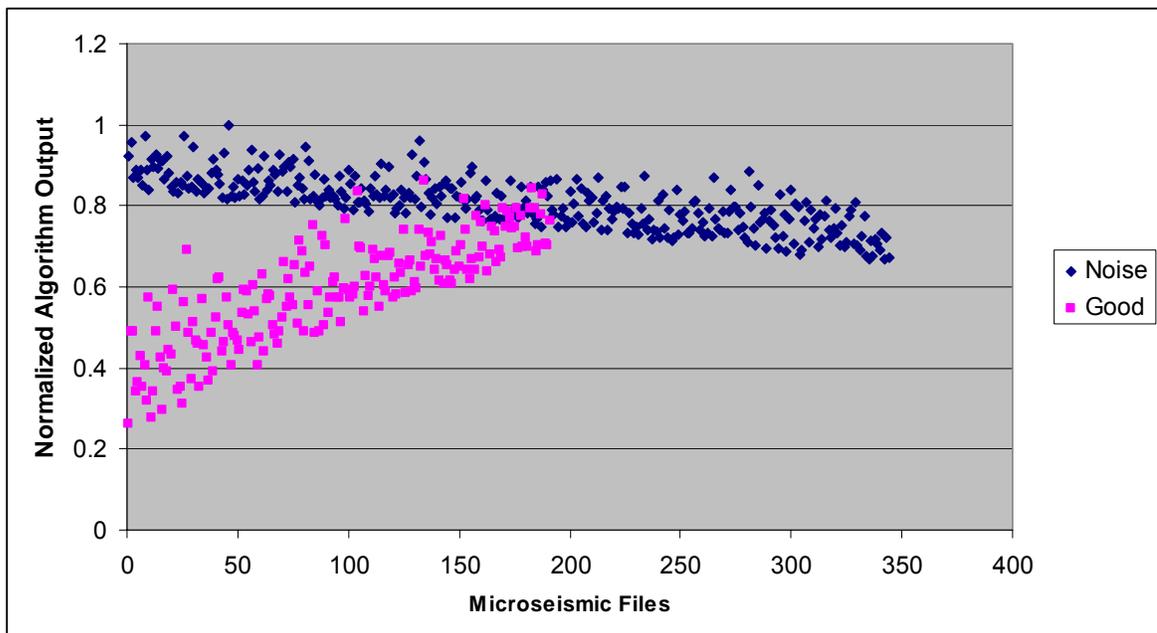


Figure 5.4: Normalized algorithm output of STA / LTA event-length detection algorithm described in section 3.2. Each pink square (blue diamond) pertains to a normalized event-length measurement for a single “good” (noise) microseismic file.

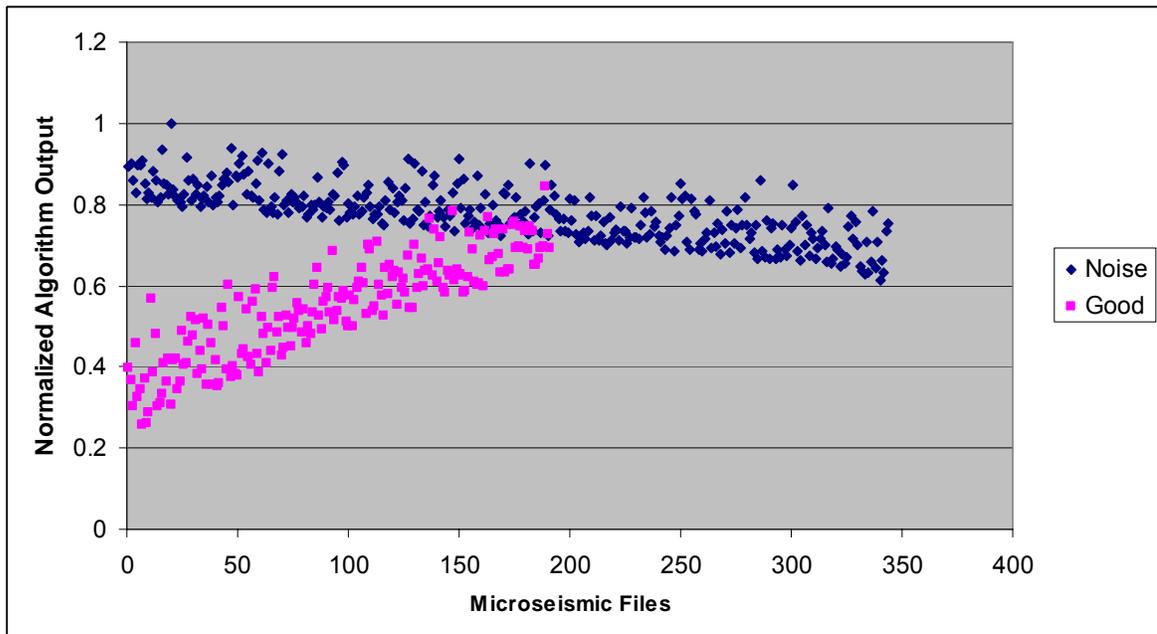


Figure 5.5: Normalized algorithm output of time-localized frequency transform event-length detection algorithm described in section 3.3. Each pink square (blue diamond) pertains to a normalized event-length measurement for a single “good” (noise) microseismic file.

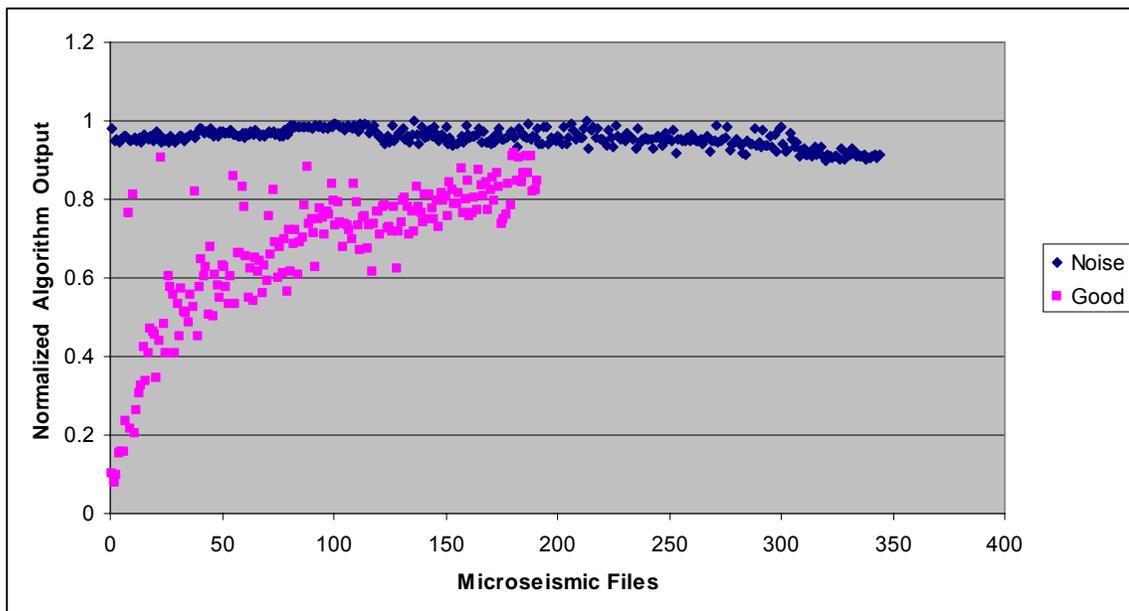


Figure 5.6: Normalized algorithm output of Threshold statistical analysis algorithm described in section 4.3.1. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.

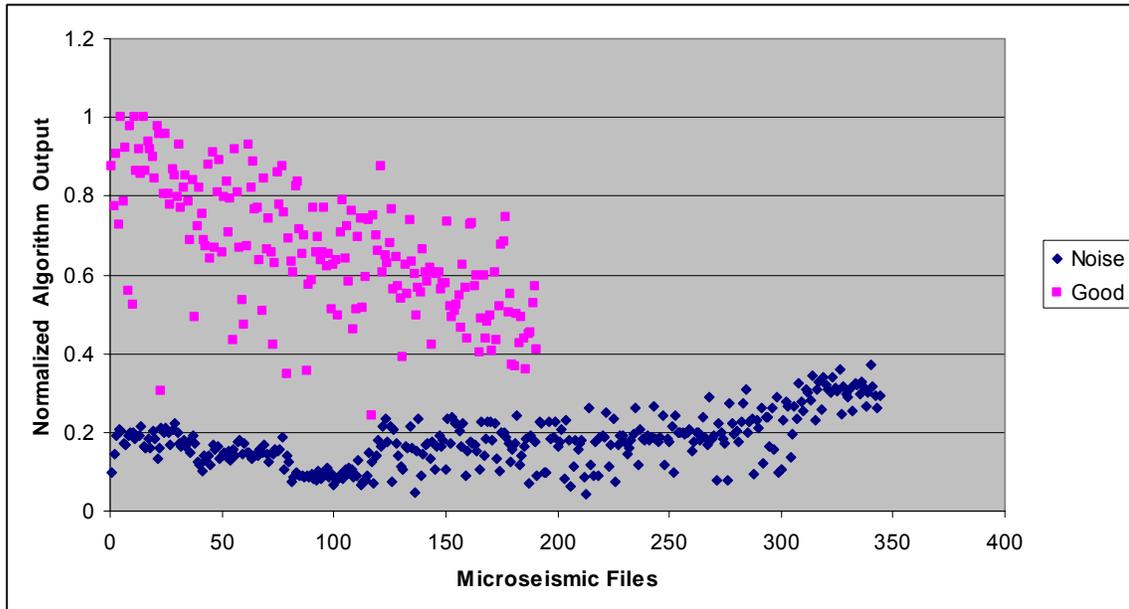


Figure 5.7: Normalized algorithm output of Histogram statistical analysis algorithm described in section 4.3.2. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.

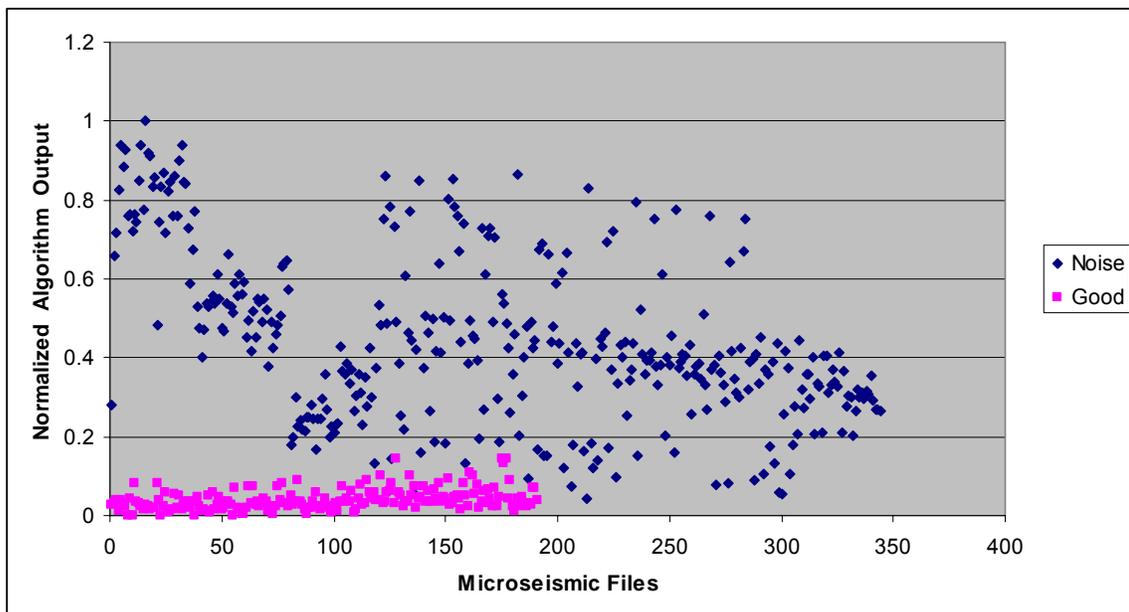


Figure 5.8: Normalized algorithm output of Specialized Zero-Crossing Count statistical analysis algorithm described in section 4.3.3. Each pink square (blue diamond) pertains to a normalized measurement for a single “good” (noise) microseismic file.

## 5.4 Applying PCA to Microseismic File Classification

### 5.4.1 Applying PCA to All Eight Algorithm Outputs

Referring to Figures 5.1 to 5.8 and equation C.1 in Appendix C, let the matrix  $\mathbf{A}$  contain algorithm measurements after normalization, where each row of  $\mathbf{A}$  pertains to a single algorithm, and each column corresponds to the algorithm outputs for a single microseismic file. Thus,  $\mathbf{A}$  will be a  $8 \times 540$  matrix. Normalization indicates that each row of  $\mathbf{A}$  has been divided by the largest value (in magnitude) found in that row. Assume that the normalized results from the low-pass filter, high-pass filter, band-pass filter, STA/LTA, time-localized frequency transform, Threshold, Histogram, and Specialized Zero-Crossing Count algorithms are placed in the first, second, third, fourth, fifth, sixth, seventh, and eighth rows of  $\mathbf{A}$ , respectively. Microseismic event files contain either 15 or 24 traces. Assume that each row of  $\mathbf{A}$  contains the normalized average algorithm output corresponding to the three traces in the microseismic file that have the strongest “good” characteristics. For example, the element  $a_{65}$  would represent the normalized average fraction of data points lying outside a predefined window (Threshold algorithm) corresponding to the three channels in the fifth microseismic file that contain the lowest fraction of outlying points.

Let  $c_{ij}$  ( $i, j = 1, 2, 3, \dots, 8$ ) represent the calculated covariance between any two of the eight algorithm outputs. For example,  $c_{26}$  represents the covariance between the normalized high-pass filter and Threshold algorithm outputs since the high-pass filter results are stored in the second row of  $\mathbf{A}$ , while the Threshold results are stored in the sixth row of  $\mathbf{A}$ . If  $i = j$ , the element  $c_{ij}$  would correspond to the calculated variance of a single

algorithm output. For example  $c_{77}$  represents the variance of the Histogram algorithm outputs.

Referring to Appendix C, the symmetric covariance matrix  $\mathbf{C}$  of the dataset can be defined as

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} & c_{17} & c_{18} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} & c_{27} & c_{28} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} & c_{36} & c_{37} & c_{38} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} & c_{46} & c_{47} & c_{48} \\ c_{51} & c_{52} & c_{53} & c_{54} & c_{55} & c_{56} & c_{57} & c_{58} \\ c_{61} & c_{62} & c_{63} & c_{64} & c_{65} & c_{66} & c_{67} & c_{68} \\ c_{71} & c_{72} & c_{73} & c_{74} & c_{75} & c_{76} & c_{77} & c_{78} \\ c_{81} & c_{82} & c_{83} & c_{84} & c_{85} & c_{86} & c_{87} & c_{88} \end{bmatrix}. \quad (5.1)$$

Applying equations C.2 to C.8, the elements in this covariance matrix were found to be

$$\mathbf{C} = \begin{bmatrix} 0.0275 & -0.0258 & -0.0261 & -0.0184 & -0.0195 & -0.0213 & 0.0338 & -0.0302 \\ -0.0258 & 0.0331 & 0.0296 & 0.0204 & 0.0218 & 0.0233 & -0.0371 & 0.0332 \\ -0.0261 & 0.0296 & 0.0349 & 0.0214 & 0.0225 & 0.0244 & -0.0381 & 0.0340 \\ -0.0184 & 0.0204 & 0.0214 & 0.0186 & 0.0172 & 0.0198 & -0.0297 & 0.0236 \\ -0.0195 & 0.0218 & 0.0225 & 0.0172 & 0.0207 & 0.0213 & -0.0313 & 0.0246 \\ -0.0213 & 0.0233 & 0.0244 & 0.0198 & 0.0213 & 0.0298 & -0.0404 & 0.0253 \\ 0.0338 & -0.0371 & -0.0381 & -0.0297 & -0.0313 & -0.0404 & 0.0667 & -0.0418 \\ -0.0302 & 0.0332 & 0.0340 & 0.0236 & 0.0246 & 0.0253 & -0.0418 & 0.0658 \end{bmatrix}. \quad (5.2)$$

The elements in  $\mathbf{C}$  confirm the general trends seen in Figures 5.1 to 5.8. For example, a general increasing trend of the pink squares in Figure 5.3 (band-pass filter algorithm) is seen with a general increasing trend of the pink squares in Figure 5.6 (Threshold algorithm), confirming  $c_{36} > 0$ . As another example, a general decreasing trend of the pink squares in Figure 5.1 (low-pass filter algorithm) is seen with a general

increasing trend in Figure 5.2 (high-pass filter algorithm), confirming  $c_{12} < 0$ . Similar arguments can be made for the trends of the blue diamonds in the figures.

Referring to Appendix C, the calculated unit-length orthogonal eigenvectors of the covariance matrix  $\mathbf{C}$ , or equivalently the principal components of the dataset  $\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2, \dots, \hat{\mathbf{E}}_8$  (equation C.18) can be calculated and stored in the transformation matrix  $\mathbf{V}$  (equation C.22). The corresponding eigenvalues ( $\lambda_1, \lambda_2, \dots, \lambda_8$ ), which are equal to data variance when projected onto the corresponding eigenvectors ( $\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2, \dots, \hat{\mathbf{E}}_8$ ) are

$$\begin{aligned}\lambda_1 &= 0.2163, \\ \lambda_2 &= 0.0294, \\ \lambda_3 &= 0.0124, \\ \lambda_4 &= 0.0053, \\ \lambda_5 &= 0.0046, \\ \lambda_6 &= 0.0042, \\ \lambda_7 &= 0.0027, \text{ and} \\ \lambda_8 &= 0.0023.\end{aligned}\tag{5.3}$$

By determining the quantity

$$\frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \lambda_6 + \lambda_7 + \lambda_8} * 100\% = 78\%,\tag{5.4}$$

it is clear that 78% of all variance present in this test dataset can be seen by projecting the data onto the first principal component  $\hat{\mathbf{E}}_1$ . Thus, for this dataset, principal components analysis has reduced effective data dimensionality from eight to one.

Figures 5.9, to 5.16 show the normalized dataset after projection onto the first through eighth principal components. Thus, referring to equation C.11 in Appendix C, Figures 5.9 to 5.16 each show data from a single row of the matrix  $\mathbf{P}$  that contains the projected data. In this specific example,  $\mathbf{P}$  is a  $8 \times 540$  matrix that contains data from the

normalized and mean-corrected matrix  $\mathbf{B}$ , also a  $8 \times 540$  matrix, after projection onto the eight principal components contained in the rows of  $\mathbf{V}$ , a square  $8 \times 8$  matrix. For this example,  $\mathbf{B}$  in equation C.11 is found with equation C.2, and  $\mathbf{V}$  is determined using equation C.22.

In Figure 5.9, a reduction in vertical overlap and improved vertical separation can be seen between the “good” and noise data points when compared with most individual algorithm results. Figure 5.10 to 5.16 appear to be noise components, and no useful information can be extracted from these figures. While some overlap has been reduced in Figure 5.9, there is still significant vertical overlap remaining after principal components analysis. Section 5.4.2 improves upon this result by restricting PCA to statistical algorithm outputs.

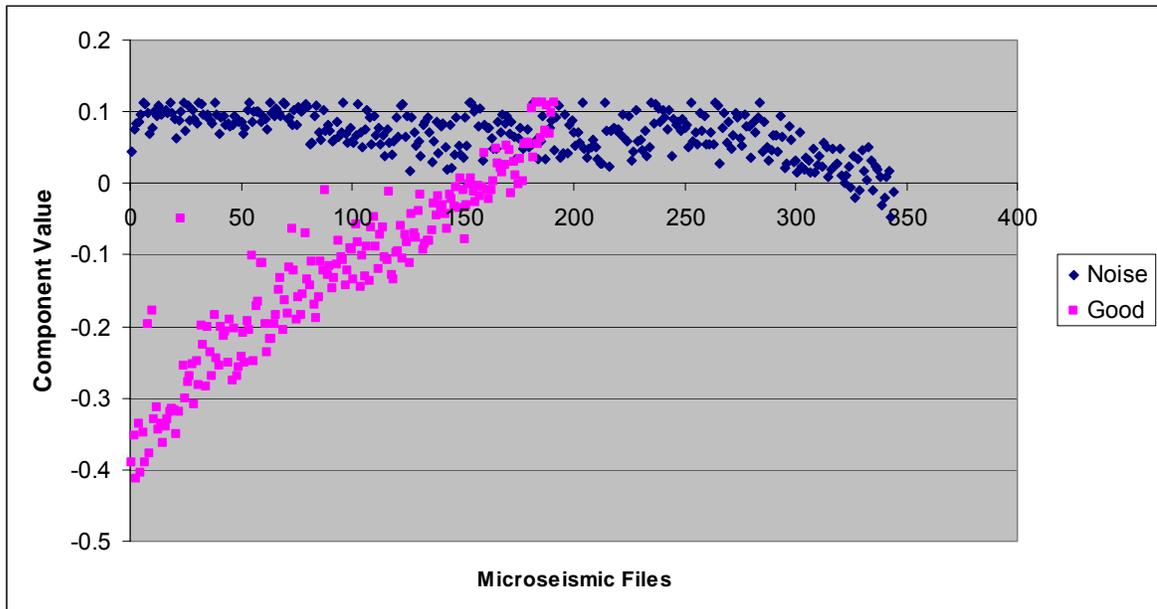


Figure 5.9: Eight-dimensional dataset projected onto first principal component. Reduction in vertical overlap is seen, but significant overlap still remains.

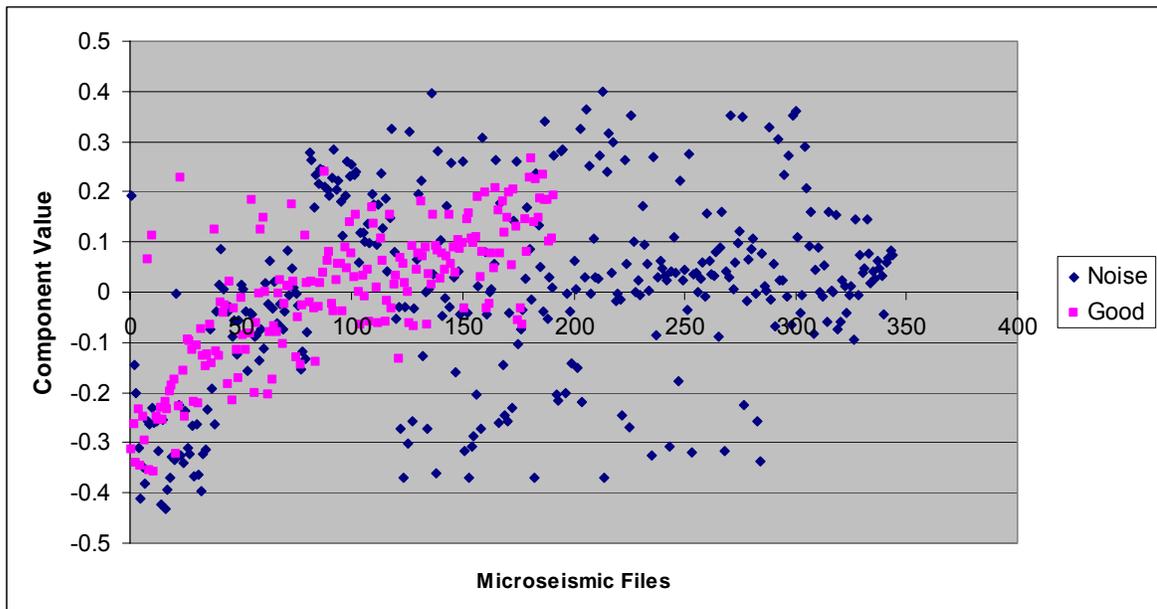


Figure 5.10: Eight-dimensional dataset projected onto second principal component. No useful information can be extracted, as only noise is seen.

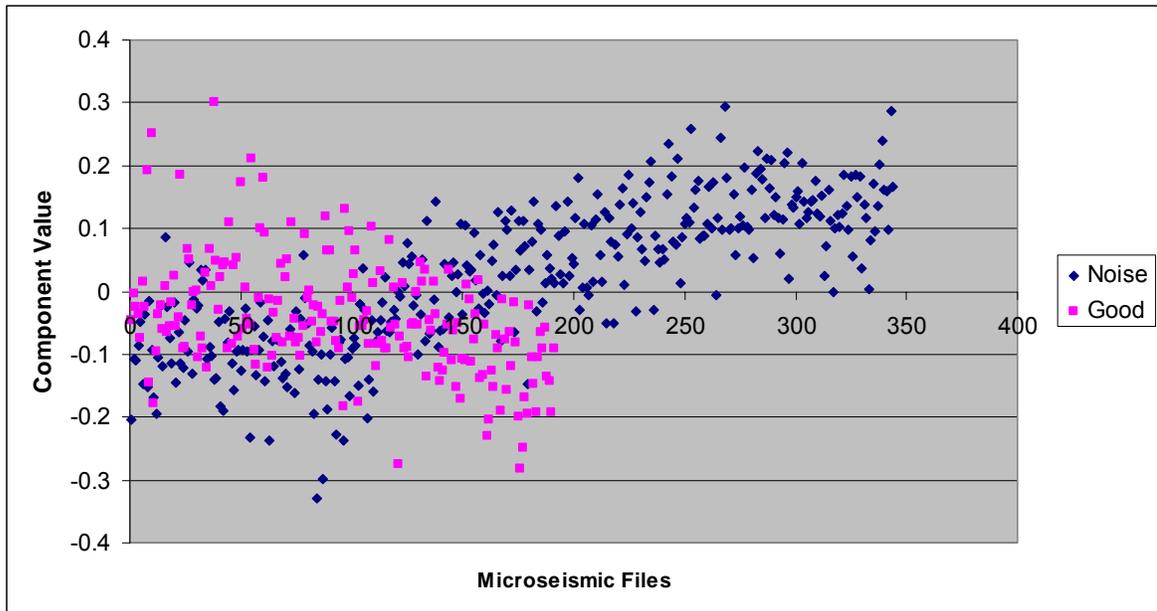


Figure 5.11: Eight-dimensional dataset projected onto third principal component. No useful information can be extracted, as only noise is seen.

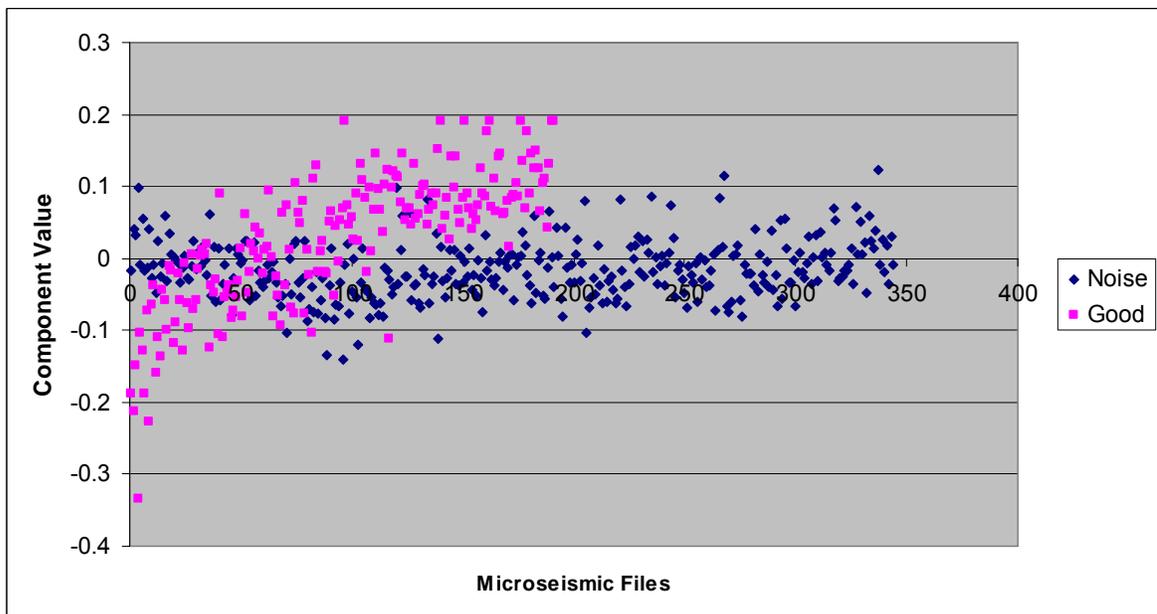


Figure 5.12: Eight-dimensional dataset projected onto fourth principal component. No useful information can be extracted, as only noise is seen.

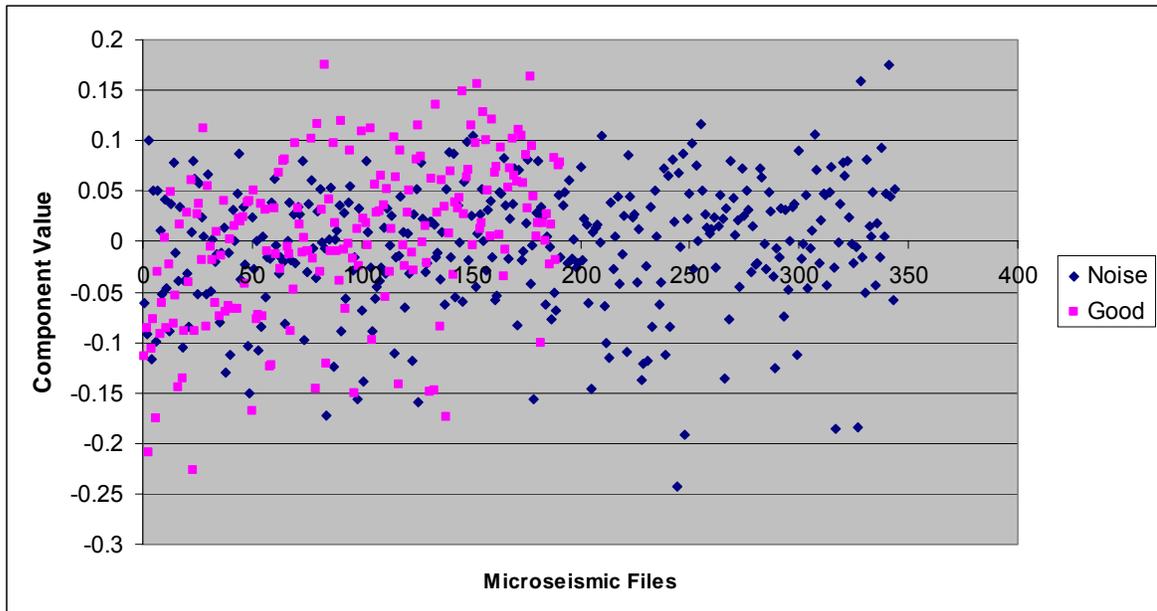


Figure 5.13: Eight-dimensional dataset projected onto fifth principal component. No useful information can be extracted, as only noise is seen.

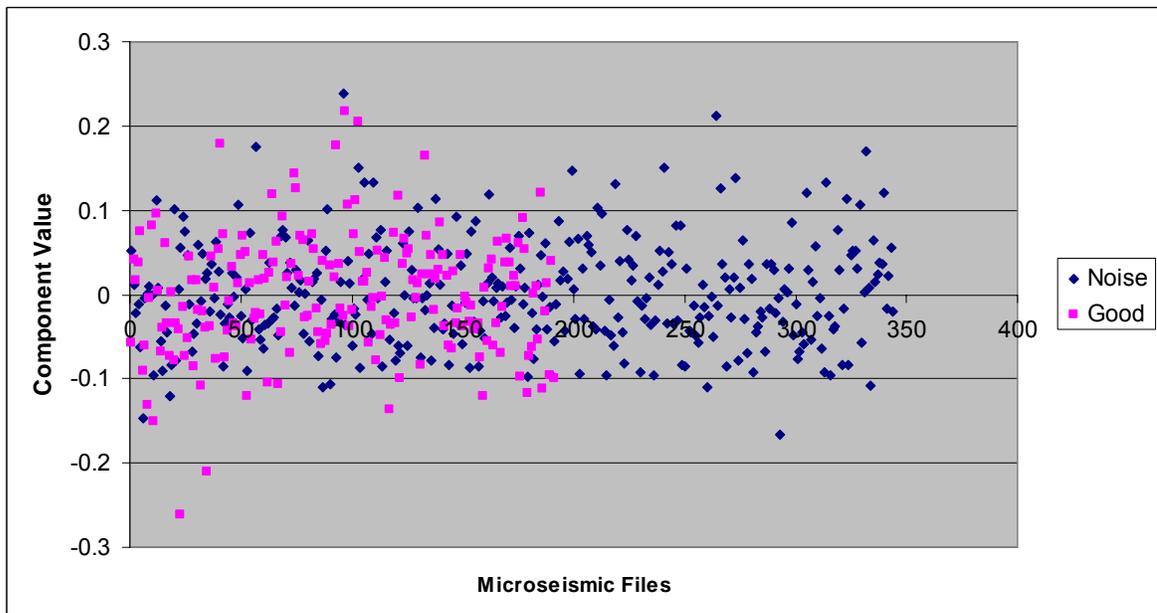


Figure 5.14: Eight-dimensional dataset projected onto sixth principal component. No useful information can be extracted, as only noise is seen.

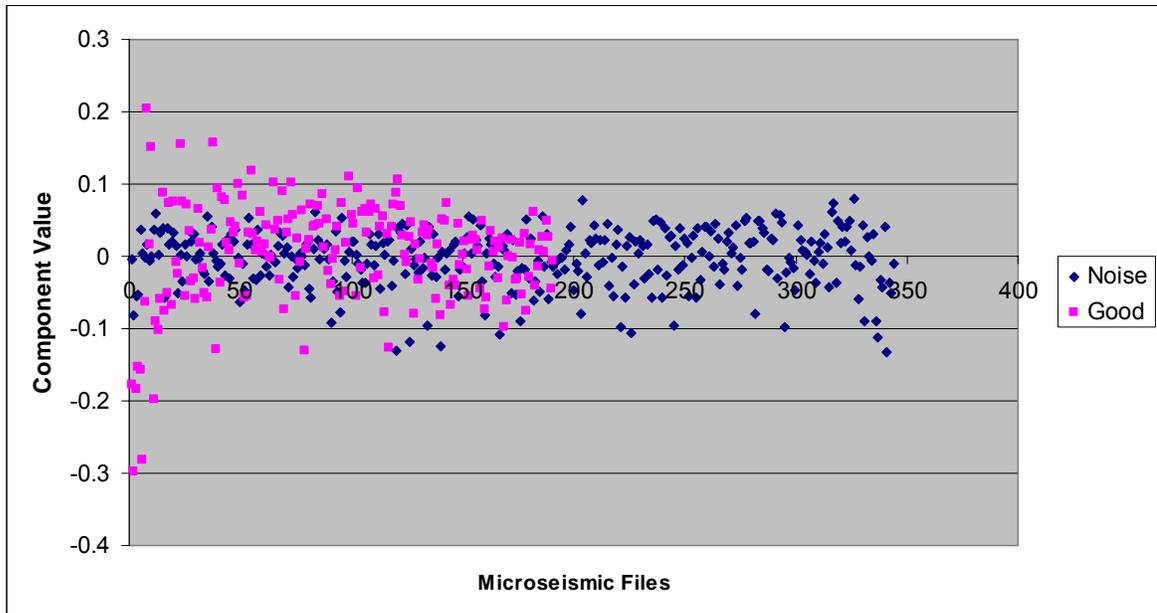


Figure 5.15: Eight-dimensional dataset projected onto seventh principal component. No useful information can be extracted, as only noise is seen.

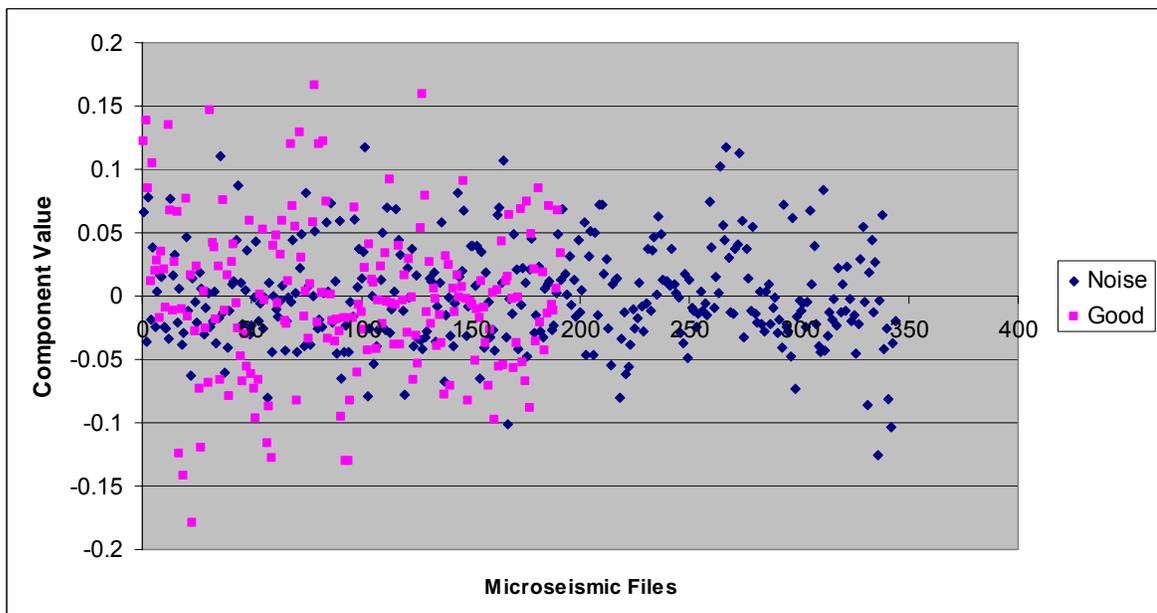


Figure 5.16: Eight-dimensional dataset projected onto eighth principal component. No useful information can be extracted, as only noise is seen.

#### 5.4.2 Restricting PCA to Statistical Algorithm Outputs

PCA is restricted to the three statistical algorithm outputs in this section to improve classification accuracy.

Referring to Figures 5.6 to 5.8 and equation C.1 in Appendix C, let the matrix  $\mathbf{A}$  contain algorithm measurements after normalization, where each row of  $\mathbf{A}$  pertains to a single statistical analysis algorithm, and each column corresponds to the algorithm outputs for a single microseismic file. Thus,  $\mathbf{A}$  will be a  $3 \times 540$  matrix. Normalization indicates that each row of  $\mathbf{A}$  has been divided by the largest value (in magnitude) found in that row. Assume that the normalized results from the Threshold, Histogram, and Specialized Zero-Crossing Count algorithms are placed in the first, second and third rows of  $\mathbf{A}$ , respectively. Microseismic event files contain either 15 or 24 traces. Assume that each row of  $\mathbf{A}$  contains the normalized average algorithm output corresponding to the three traces in the microseismic file that have the strongest “good” characteristics. For example, the element  $a_{15}$  would represent the normalized average fraction of data points lying outside a predefined window (Threshold algorithm) corresponding to the three channels in the fifth microseismic file that contain the lowest fraction of outlying points.

Let  $c_{TT}$ ,  $c_{HH}$ , and  $c_{ZZ}$  represent the calculated variance of the normalized Threshold, Histogram, and Specialized Zero-Crossing Count algorithm outputs, respectively. Also let  $c_{TH}$ ,  $c_{TZ}$ , and  $c_{HZ}$  represent the calculated covariance between the normalized Threshold-Histogram, Threshold-Zero Crossing Count, and Histogram-Zero Crossing Count algorithm outputs, respectively. Applying equations C.2 to C.8 in Appendix C, the symmetric covariance matrix  $\mathbf{C}$  of the dataset was found to be

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} = \begin{bmatrix} c_{TT} & c_{TH} & c_{TZ} \\ c_{TH} & c_{HH} & c_{HZ} \\ c_{TZ} & c_{HZ} & c_{ZZ} \end{bmatrix} = \begin{bmatrix} 0.0298 & -0.0404 & 0.0253 \\ -0.0404 & 0.0667 & -0.0418 \\ 0.0253 & -0.0418 & 0.0658 \end{bmatrix}. \quad (5.5)$$

The elements in  $\mathbf{C}$  confirm the general trends seen in Figures 5.6, 5.7, and 5.8. For example, a general increasing trend of the pink squares in Figure 5.6 (Threshold algorithm) is seen with a general decreasing trend of the pink squares in Figure 5.7 (Histogram algorithm), confirming  $c_{TH} < 0$ . A similar argument can be made for the trends of the blue diamonds in the figures.

The calculated unit-length orthogonal eigenvectors of the covariance matrix  $\mathbf{C}$ , or equivalently the principal components of the dataset ( $\hat{\mathbf{E}}_1$ ,  $\hat{\mathbf{E}}_2$ , and  $\hat{\mathbf{E}}_3$ ), found using equation C.18, are stored in the first, second, and third columns of a square matrix  $\mathbf{F}$ . For this example test dataset,

$$\mathbf{F} = \begin{bmatrix} 0.4246 & -0.3624 & -0.8297 \\ -0.6725 & 0.4873 & -0.5570 \\ 0.6062 & 0.7945 & -0.0368 \end{bmatrix}. \quad (5.6)$$

The eigenvalues ( $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$ ), which are equal to data variance when projected onto the corresponding eigenvectors ( $\hat{\mathbf{E}}_1$ ,  $\hat{\mathbf{E}}_2$ , and  $\hat{\mathbf{E}}_3$ ) are

$$\begin{aligned} \lambda_1 &= 0.1298, \\ \lambda_2 &= 0.0287, \text{ and} \\ \lambda_3 &= 0.0038. \end{aligned} \quad (5.7)$$

By determining the quantity

$$\frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} * 100\% = 80\%, \quad (5.8)$$

it is clear that 80% of all variance present in this test dataset can be seen by projecting the data onto the first principal component  $\hat{\mathbf{E}}_1$ .

Figures 5.17, 5.18, and 5.19 show the normalized dataset after projection onto the first, second, and third principal components, respectively. Thus, referring to equation C.11, Figures 5.17, 5.18, and 5.19 show data from the first, second, and third rows of the matrix  $\mathbf{P}$ , respectively. In this specific example,  $\mathbf{P}$  is a  $3 \times 540$  matrix that contains data from the normalized and mean-corrected matrix  $\mathbf{B}$ , also a  $3 \times 540$  matrix, after projection onto the three principal components contained in the rows of  $\mathbf{V}$ , a square  $3 \times 3$  matrix. For this example,  $\mathbf{B}$  in equation C.11 is found with equation C.2, and  $\mathbf{V}$  is determined using equation C.22.

In Figure 5.17, no vertical overlap exists between the “good” and noise data points, which clearly suggests that file classification using Figure 5.17 corresponding to PCA would be improved over attempting to empirically classify files with the normalized raw measurements shown in Figures 5.6 to 5.8. The results shown in Figure 5.17 are a significant improvement over Figure 5.9, where all eight algorithm outputs were used. This improvement is seen through improved vertical separation and the complete removal of vertical overlap. Figures 5.18 and 5.19 correspond to noise components in the data and are not useful for microseismic file classification. Thus, for this dataset, PCA has reduced the effective dimensionality from three to one.

Figure 5.17 suggests that if PCA is used for multivariate data reduction, all 540 files from 28 different pads in this specific example dataset could be classified to perfect accuracy, which is encouraging. This obviously will not always be the case for all datasets, but PCA does allow an analytical examination of multivariate classification measurements

to determine the “direction” where the most important classification information resides. This technique results in improved classification accuracy over simply observing individual algorithm outputs for classification. Restricting PCA to statistical analysis algorithm outputs further improves classification accuracy.

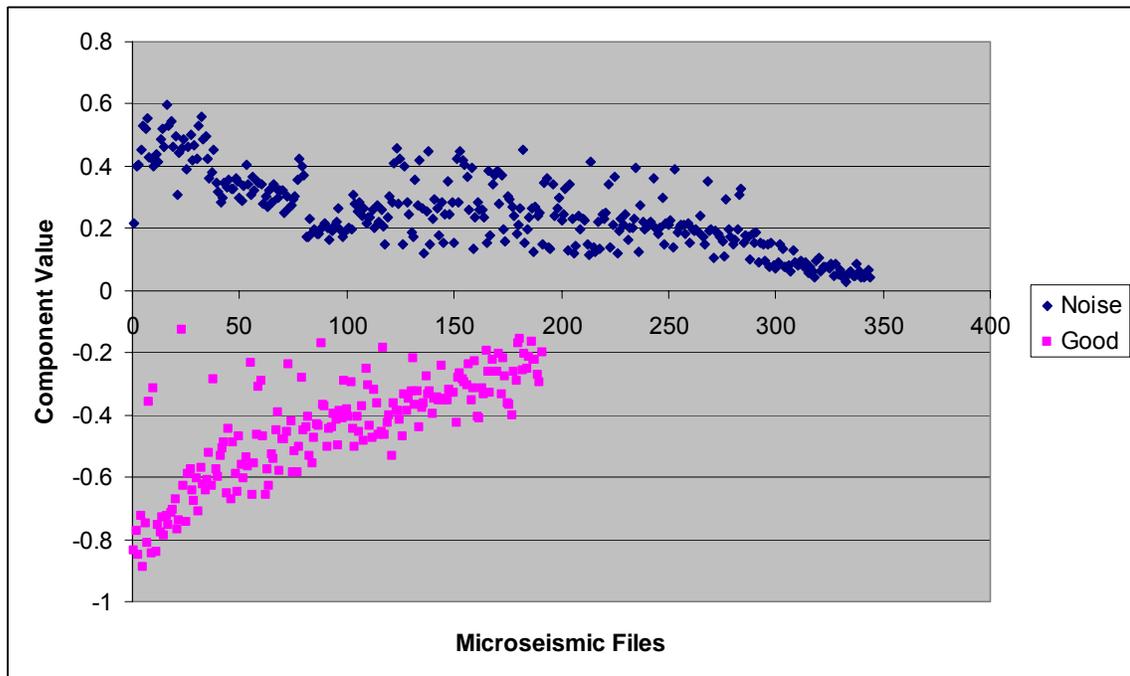


Figure 5.17: Three-dimensional dataset projected onto first principal component. No vertical overlap is seen between “good” and noise points. For this specific test dataset, perfect classification accuracy could be achieved using PCA and examining this component.

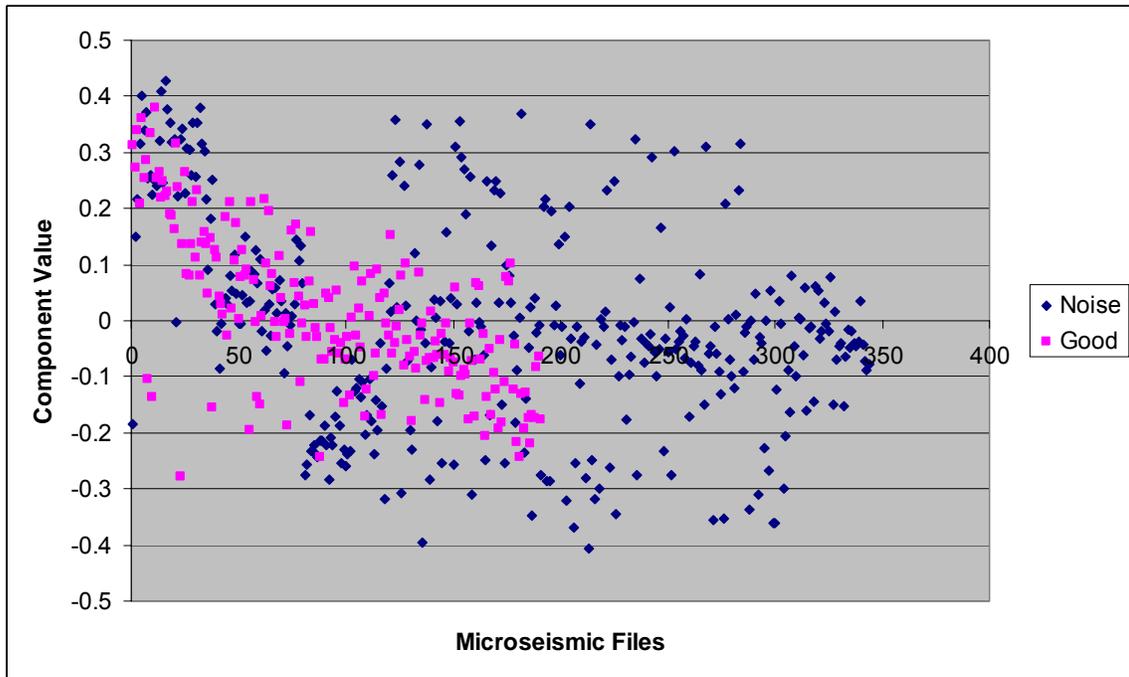


Figure 5.18: Three-dimensional dataset projected onto second principal component. No useful information can be extracted, as only noise is seen.

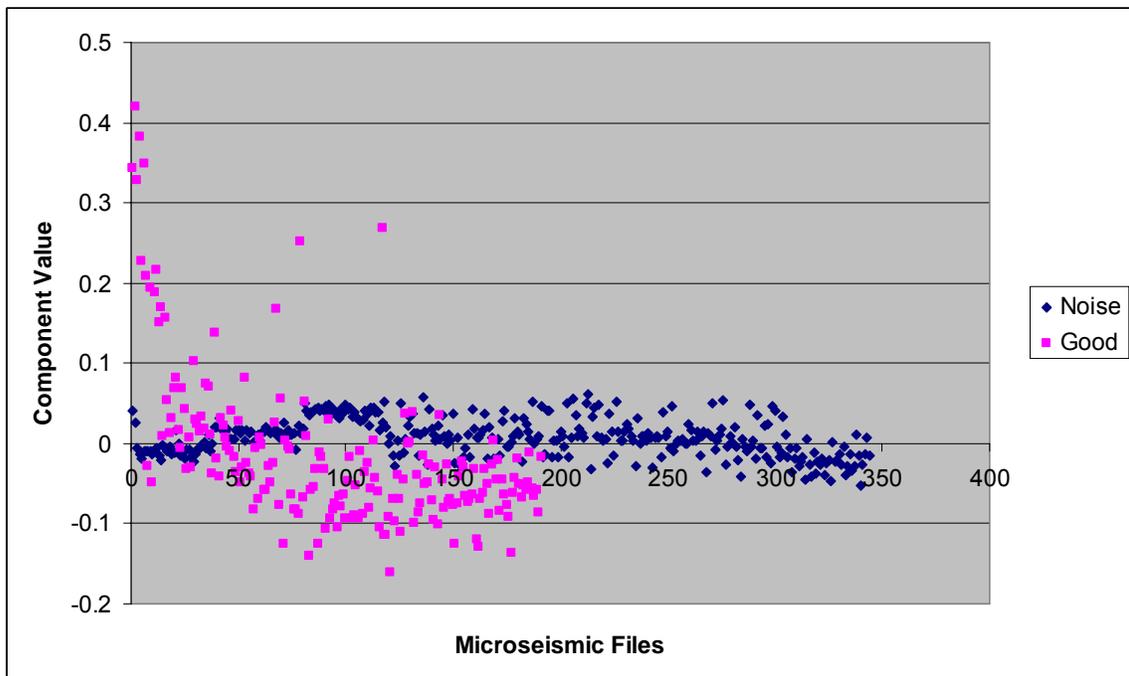


Figure 5.19: Three-dimensional dataset projected onto third principal component. No useful information can be extracted, as only noise is seen.

## 5.5 Conclusion

Testing shown in section 5.3 suggests that statistical analysis algorithms discussed in chapter four yield results with the highest potential for maximally accurate microseismic file classification. Principal components analysis, a multivariate data reduction technique, was first applied to all eight algorithm outputs pertaining to a diverse test dataset of 540 microseismic files from 28 different production pads. Following this, principal components analysis was applied only to statistical algorithm outputs, which resulted in improved classification accuracy.

For this specific dataset, applying PCA to statistical algorithm outputs would result in perfect microseismic file classification. This perfect classification result is not expected over all datasets, but the demonstrated improvement after applying PCA in microseismic file classification is expected.

## **CHAPTER SIX: Algorithm Implementation**

### **6.1 Introduction**

The microseismic signal analysis techniques described in chapters 2 to 5 were implemented in MATLAB<sup>®</sup>. Two MATLAB<sup>®</sup> program implementations were realized. For the first implementation, a graphical user interface (GUI) was created that applied the discussed algorithms in an equally-weighted fashion for microseismic file classification. Following the first implementation, extensive testing demonstrated that statistical analysis techniques discussed in chapter 4 yield results with the highest potential for consistently accurate microseismic file classification. These statistical algorithms gave the most consistent clustered results when tested on “good” and noise files. Following these tests, a condensed classification function pertaining to a second, different, implementation was created. This function applied principal components analysis (PCA) to statistical algorithm outputs for microseismic file classification and yielded improved results over the first implementation.

Results from the first implementation were encouraging for a very specific dataset, as algorithm settings were optimized to this dataset. These results, however, did not carry over to other, more diverse, datasets.

The results, detailed in section 6.3, from the second implementation are encouraging over a wide range of datasets. This implementation has demonstrated to be robust, as function settings can be left unchanged when applied to various datasets. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this developed application could have significant future impact.

Section 6.2 provides general implementation outline examples for illustrative purposes. The implemented programs were similar, but not identical to, the descriptions in section 6.2.

## 6.2 Program Implementations

### 6.2.1 Graphical User Interface

For the first implementation, the discussed algorithms were combined and optimized to a preliminary test dataset of microseismic event files, most of which were from only 5 different production pads. A graphical user interface (GUI) application was created in MATLAB<sup>®</sup>. This application, entitled "Event\_Analyzer", classifies and separates microseismic event files into noise events and "good" events. Figure 6.1 depicts the appearance of this GUI upon program startup.

In Figure 6.1, the algorithms that influence the microseismic file classification appear in the "Decision Settings" panel with the default deciding thresholds shown below the labels. *LPF*, *HPF*, and *BPF* correspond to the low-pass filtering, high-pass filtering, and band-pass filtering algorithms described in chapter 2, respectively. *FDM* corresponds to the frequency-domain event-length detection algorithm described in chapter 3. *Thresh*, *Hist*, and *SR* pertain to the "Threshold", "Histogram", and "Specialized Zero-Crossing Count" statistical analysis algorithms described in chapter 4, respectively. When the GUI runs, the results from these seven algorithms determine whether a file is classified as "good" or noise. The time-domain event-length detection (STA/LTA) method was not implemented in the GUI because it significantly increased the program runtime. Also, a technique similar to

this STA / LTA method was already previously applied to the received test data in order to detect events and to limit microseismic traces to 1.365 (or 1.5) seconds of activity.

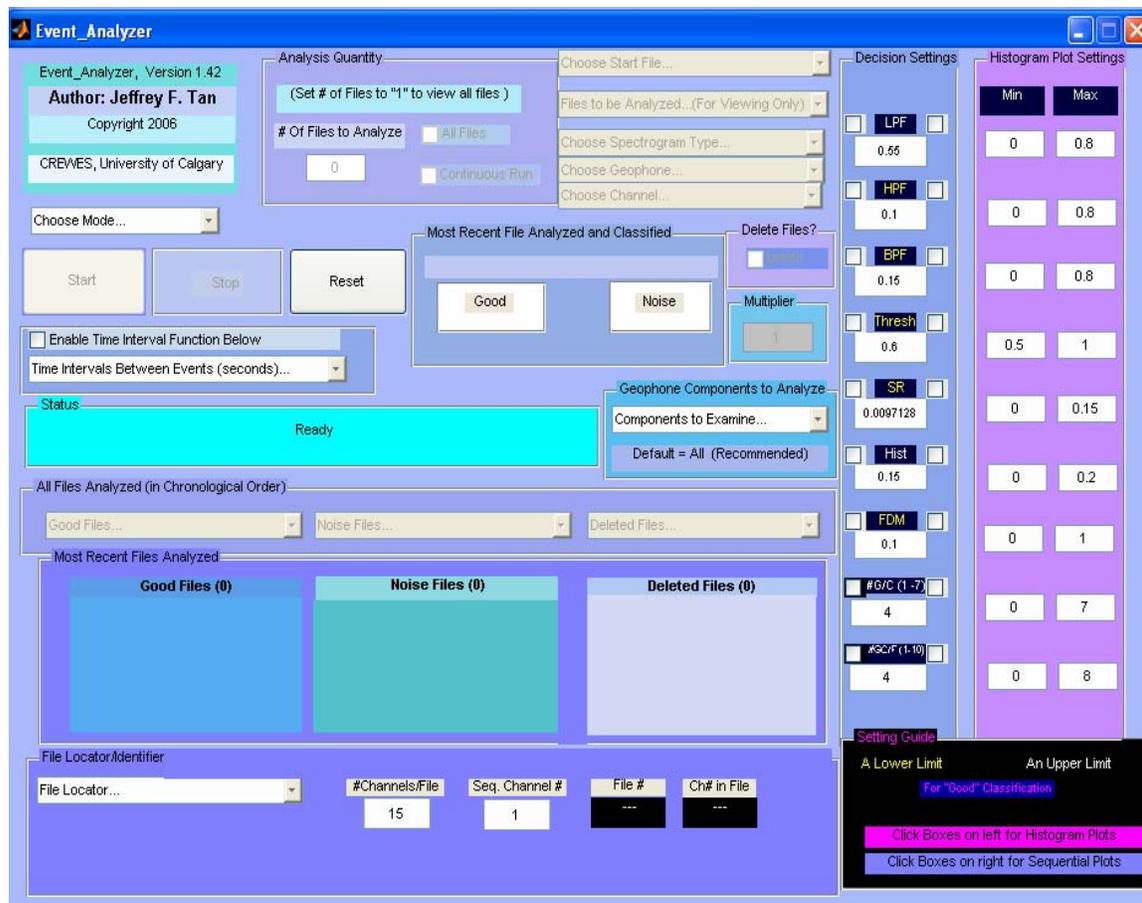


Figure 6.1: “Event\_Analyzer” program upon startup.

During program run, each microseismic signal analysis algorithm is applied to each of the traces in an event file. Results from the algorithms are coded as "1" if the algorithm deems the trace to contain a "good" event, and "0" otherwise. The algorithm results are summed on a per-trace basis providing a per-trace score. Traces that have a score exceeding a threshold are flagged as "good", while the remaining traces are flagged as "noise". The overall file classification is determined by identifying the total number of "good" traces (Tan et al., 2006).

As an example, the "good" and noise traces shown in Figures 1.7 and 1.17, respectively, are passed through the GUI. Assume that a trace will be classified as "good" if it "passes" at least four of the seven algorithms. This setting is adjustable under the #G/C(I-7) label in the "Decision Settings" panel in Figure 6.1. A given trace "passes" an algorithm if that algorithm's output falls within a predetermined range. For the frequency-filtering techniques (*LPF*, *HPF*, and *BPF*), the algorithm outputs correspond to the peak amplitude of the trace after filtering. The algorithm output of the frequency-domain event-length detection technique (*FDM*) is the calculated trace event-length. The algorithm outputs of the statistical analysis techniques (*Thresh*, *Hist*, and *SR*) pertain to the fraction of outlying data points (*Thresh*), the fraction of data points falling in the range pertaining to a centre histogram bin (*Hist*), and the fraction of polarity reversals after low-amplitude noise is removed (*SR*). Table 6.1 shows how the "good" and noise example traces in Figures 1.7 and 1.17 perform for each algorithm.

Table 6.1: Performance of example traces. Trace values that "pass" are in boldface.

Algorithm	Passing Range	"Good" Trace Value	Noise Trace Value
LPF	$\geq 0.55$	0.49	0.131
HPF	$\leq 0.1$	<b>0.0484</b>	0.708
BPF	$\leq 0.15$	0.165	0.364
Thresh	$\leq 0.6$	<b>0.208</b>	0.682
SR	$\leq 0.0097128$	<b>0.00024414</b>	0.0728
Hist	$\geq 0.15$	<b>0.346</b>	0.107
FDM (seconds)	$\leq 0.1$	<b>0.040333</b>	0.534

Since the example "good" trace from Figure 1.7 passes five algorithms, this trace would be classified "good". The example noise trace passes none of the algorithms and would thus be classified as noise. Microseismic event files contain either 15 or 24 traces.

This analysis repeats for every trace in a file, and based on the number of "good" traces, the file is classified as either "good" or noise. A "good" file classification could require that there are at least four "good" traces in an event file, for example. This default setting is adjustable by changing the  $\#GC/F(I-10)$  label in the "Decision Settings" panel in Figure 6.1. After a microseismic file is classified, the program automatically proceeds to the next file in the queue, and this procedure is repeated until all files have been classified.

### 6.2.2 Principal Components Analysis Applied

Testing that followed the GUI implementation described in section 6.2.1 demonstrated that the statistical analysis techniques yielded results capable of more robust and accurate microseismic file classification. The statistical analysis techniques were also found to be the most computationally efficient, a required characteristic when potentially classifying tens of thousands of files daily. The second implementation applied principal components analysis (PCA), described in chapter 5 and Appendix C, to statistical algorithm outputs for file classification.

First, it is required to obtain the principal components from a reference dataset. To ensure classification robustness, this dataset should be as diverse as possible. Thus, it is best to obtain data from as many different production pads as possible. The statistical algorithm outputs pertaining to an incoming microseismic file can then be projected onto the principal components obtained from the reference dataset for classification.

Assume the reference dataset contains  $n$  microseismic files. Referring to equation C.1, let  $\mathbf{A}$  be a  $3 \times n$  matrix containing normalized statistical analysis algorithm outputs. Normalization indicates that each row of  $\mathbf{A}$  has been divided by the largest value (in

magnitude) found in that row. Let  $max_i$  denote the maximum algorithm output value over all dataset files pertaining to the Threshold ( $i = 1$ ), Histogram ( $i = 2$ ), and Specialized Zero-Crossing Count ( $i = 3$ ) algorithms, before normalization. These quantities will be required at a later stage. The rows of  $\mathbf{A}$  pertain to different normalized algorithm outputs, while its columns correspond to different microseismic files. Assume that the normalized results from the Threshold, Histogram, and Specialized Zero-Crossing Count algorithms are placed in the first, second, and third rows of  $\mathbf{A}$ , respectively. Microseismic event files contain either 15 or 24 traces. Assume that each row of  $\mathbf{A}$  contains the normalized average algorithm output corresponding to the three traces in the microseismic file that have the strongest “good” characteristics. For example, the element  $a_{15}$  would represent the normalized average fraction of data points lying outside a predefined window (Threshold algorithm) corresponding to the three channels in the fifth microseismic file that contain the lowest fraction of outlying points.

Referring to equation C.2, let  $\mathbf{B}$  represent the reference dataset matrix  $\mathbf{A}$  after row means have been subtracted. Let  $\mu_i$  denote the row means pertaining to the normalized Threshold ( $i = 1$ ), Histogram ( $i = 2$ ), and Specialized Zero-Crossing Count ( $i = 3$ ) algorithm outputs in  $\mathbf{A}$ . These values will be required at a later stage. Applying equation C.8, the covariance matrix  $\mathbf{C}$ , pertaining to the data in  $\mathbf{B}$ , can be calculated. Following this, the unit-length eigenvectors of  $\mathbf{C}$  ( $\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2, \hat{\mathbf{E}}_3$ ) can be found, which are equivalent to the principal components of the dataset. Subsequently, each principal component can be stored in a column of a matrix  $\mathbf{F}$ .

The algorithm outputs stored in  $\mathbf{B}$  can then be projected onto the unit-length eigenvectors, to obtain projected data (stored in  $\mathbf{P}$ ), by applying equations C.11 and C.22.

The first, second, and third rows of  $\mathbf{P}$ , pertaining to data projected onto the first, second, and third principal components, can then be examined to empirically determine classification boundaries. For example, if Figures 5.17, 5.18, and 5.19 represent the reference dataset projected onto its first, second, and third principal components, respectively, a file could be deemed as “good” (noise) if its first principal component value is less than zero (greater or equal to zero). Depending on the reference dataset, the second and third principal components could also be used for microseismic file classification.

After the principal components of a reference dataset have been calculated, a single incoming microseismic file can be classified. Let  $T$ ,  $H$ , and  $Z$  pertain to the measured Threshold, Histogram, and Specialized Zero-Crossing Count algorithm output values, respectively, for a single microseismic file to be classified. Define a 3-element column-vector  $\mathbf{B}_{\text{file}}$  as

$$\mathbf{B}_{\text{file}} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \frac{T}{\max_1} - \mu_1 \\ \frac{H}{\max_2} - \mu_2 \\ \frac{Z}{\max_3} - \mu_3 \end{bmatrix}, \quad (6.1)$$

where  $\max_i$  and  $\mu_i$  ( $i = 1, 2, 3$ ) were obtained from the reference dataset. The elements of  $\mathbf{B}_{\text{file}}$  can then be projected onto the principal components of the reference dataset contained in  $\mathbf{F}$ . This projected data can be stored in the column-vector  $\mathbf{P}_{\text{file}}$  given as

$$\mathbf{P}_{\text{file}} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \mathbf{V}\mathbf{B}_{\text{file}} = \mathbf{F}^T\mathbf{B}_{\text{file}}, \quad (6.2)$$

where  $p_i$  ( $i = 1, 2, 3$ ) represents the algorithm outputs pertaining to the single microseismic file of interest projected onto the  $i^{\text{th}}$  principal component of the reference dataset.

The elements of  $\mathbf{P}_{\text{file}}$  can then be examined to classify the microseismic file using classification boundaries previously determined with the reference dataset. For example, referring to Figure 5.17 and the preceding reference dataset discussion, a single microseismic file could be classified as “good” if  $p_1 < 0$ , and noise otherwise. Depending on the reference dataset,  $p_2$  and  $p_3$  could also be used for microseismic file classification.

Once a single microseismic file has been classified, subsequent files in the queue can be classified through application of equations 6.1 and 6.2.

### 6.3 Results

The first GUI implementation that applied the developed algorithms in an equally-weighted fashion was not adequately robust. Algorithm settings required adjustments when datasets were altered. For example, after algorithm settings were optimized to a test dataset where most microseismic files originated from five production pads, an accuracy of 99.9% was obtained; however, accuracy decreased to approximately 70% when tested on more diverse datasets.

The second implementation that applied principal components analysis to statistical algorithm outputs was found to be robust and yield accurate results over diverse datasets without altering any program settings. When applied to a specific dataset where most files originated from five production pads, an accuracy of 99.5% was obtained. Testing on a more diverse dataset with files from 28 different pads yielded a classification accuracy of

98.8%. An exhaustive test on microseismic files from 72 different production pads resulted in a 90% accuracy. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this developed application could have significant future impact.

#### **6.4 Conclusion**

Two implementation schemes were designed to apply the developed algorithms to microseismic file classification. The first was a graphical user interface (GUI) that applied the developed algorithms in an equally-weighted fashion. After this first implementation, it was found that statistical analysis algorithms yielded the most consistent clustered results when tested on “good” and noise files, suggesting that these algorithms were most capable of accurate microseismic file classification. The second implementation applied principal components analysis (PCA) to statistical algorithm outputs to classify files.

The first implementation was not found to be adequately robust, as algorithm settings required adjustments when datasets were altered. The second implementation demonstrated robustness over a wide range of datasets, yielding classification accuracies between 90% and 99.5%. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this developed application could have significant future impact.

## **CHAPTER SEVEN: Conclusions and Future Work**

### **7.1 Summary**

Passive-seismic event-classification algorithms were developed and applied to synthetic data and to microseismic events generated from Imperial Oil's passive-seismic monitoring system at Cold Lake, Alberta, where heavy-oil production is present. The purpose was to develop algorithms capable of differentiating "good" events worth further investigation, from noise. Novel algorithms were required, as the current event-file classification software has been known to misclassify a large portion of microseismic files, resulting in "good" events and noise events being incorrectly identified. Numerous misclassifications require extensive manual investigation, which can become very costly.

Based on the observation that many "good" events contain lower signal frequencies than noise events, frequency filters with practical responses were developed. A low-pass Inverse-Chebyshev filter, high-pass Butterworth filter, and band-pass Chebyshev filter were developed and applied to example "good" and noise traces. Peak amplitude differences were seen pertaining to example "good" and noise traces after filtering.

Event-length detection techniques were developed in the time- and frequency-domains based on the observation that P-wave event-lengths of many "good" traces are significantly shorter than noise event-lengths. The time-domain technique examined ratios of short-term to long-term energy averages to determine event-lengths. The frequency-domain technique examined high-frequency content present in time-localized Fourier transform windows. Calculated event-length differences were seen corresponding to example "good" and noise traces.

Statistical analysis techniques were developed based on the observations that, compared to noise, many “good” traces contain less signal variance, stronger centralized data distribution, and less sporadic sequential time-series behaviour about its mean. Computationally-efficient algorithms that examined the fraction of time-series data points outside a predefined window; determined centralized data-point concentration; and examined the fraction of polarity reversals after low amplitude noise was removed were developed. When applied to example “good” and noise traces, significant algorithm output differences were seen.

Extensive testing demonstrated that the developed statistical analysis techniques yield results with the highest potential for consistently accurate microseismic file classification, as these statistical algorithms yielded the most consistent clustered results when tested on “good” and noise files. Principal components analysis (PCA) was applied to project statistical algorithm output data onto components of maximum data variance in order to optimize microseismic file classification.

Two implementation schemes were created to apply the developed algorithms to microseismic file classification. The first applied the developed algorithms in an equally-weighted fashion through a graphical user interface (GUI). This implementation was found to be inadequately robust, as program settings required alteration over varying datasets. The second implementation applied PCA to statistical algorithm outputs. This latest implementation demonstrated robust accurate performance over a wide range of datasets, as program settings could be held constant over different datasets. Given that up to tens of thousands of microseismic events are detected daily at Cold Lake, this developed application could have significant future impact.

## 7.2 Future Work

Empirically, microseismic events can be deemed to be “good” through observation of a distinct and impulsive P-wave arrival followed by a lower-frequency time-delayed S-wave arrival. Another classification technique could be developed that characterizes a microseismic event as “good” based on a successful automated search of an impulsive P-wave arrival followed by an S-wave arrival of lower frequency.

## Appendix A

### *Inverse-Chebyshev Low-Pass Frequency Response*

This derivation follows the development in Maundy (2005).

To obtain this filter's transfer function,  $T(j\omega)$ , where  $\omega$  is the angular frequency and  $j = \sqrt{-1}$ , the magnitude-squared function is first written as

$$|T(j\omega)|^2 = \frac{A(\omega^2)}{A(\omega^2) + P(\omega^2)} = \frac{1}{1 + \frac{P(\omega^2)}{A(\omega^2)}}, \quad (\text{A.1})$$

where

$$P(\omega^2) = 1, \text{ and } A(\omega^2) = KC_n^2 \left( \frac{\omega_s}{\omega} \right). \quad (\text{A.2})$$

In equation A.2,  $K$  is a constant to be determined, and  $C_n(\cdot)$  is the  $n^{\text{th}}$  order Inverse-Chebyshev function given by

$$\begin{aligned} C_n(\Omega) &= \cosh(n \cosh^{-1}(\Omega)), & |\Omega| < 1 \\ C_n(\Omega) &= \cos(n \cos^{-1}(\Omega)), & |\Omega| \geq 1 \end{aligned} \quad \Omega = \frac{\omega}{\omega_s}, \quad (\text{A.3})$$

where  $\Omega$  represents the radial frequency  $\omega$  normalized to the stopband edge frequency  $\omega_s$ .

Substituting  $\Omega = \frac{\omega}{\omega_s}$  into equation A.2 and using this result in equation A.1 yields

$$|T(j\Omega)|^2 = \frac{1}{1 + \frac{1}{KC_n^2 \left( \frac{1}{\Omega} \right)}} = \frac{KC_n^2 \left( \frac{1}{\Omega} \right)}{1 + KC_n^2 \left( \frac{1}{\Omega} \right)}. \quad (\text{A.4})$$

In dB, the filter's amplitude response,  $T_{dB}(j\Omega)$  is given by

$$T_{dB}(j\Omega) = 20 \log(|T(j\Omega)|) = 20 \log \sqrt{\frac{1}{1 + \frac{1}{KC_n^2\left(\frac{1}{\Omega}\right)}}} = -10 \log \left( 1 + \frac{1}{KC_n^2\left(\frac{1}{\Omega}\right)} \right). \quad (\text{A.5})$$

The attenuation function in dB,  $\alpha(\Omega)$ , is

$$\alpha(\Omega) = -T_{dB}(j\Omega) = 10 \log \left( 1 + \frac{1}{KC_n^2\left(\frac{1}{\Omega}\right)} \right). \quad (\text{A.6})$$

To find  $K$ , equation A.3 is used with the substitution  $\Omega=1$  and thus

$C_n(\Omega) = C_n(1) = 1$ . Using this result in equation A.6 gives

$$\alpha(1) = -T_{dB}(j1) = 10 \log \left( 1 + \frac{1}{KC_n^2\left(\frac{1}{1}\right)} \right) = 10 \log \left( 1 + \frac{1}{K} \right) = \alpha_{\min}. \quad (\text{A.7})$$

Note that  $\alpha(1) = \alpha_{\min}$ , because  $\Omega = 1$  corresponds to a radial frequency of  $\omega_s$ . From Figure 2.1, the attenuation at  $\omega_s$  is  $\alpha_{\min}$ . Solving equation A.7 for  $K$  yields

$$K = (10^{0.1\alpha_{\min}} - 1)^{-1}. \quad (\text{A.8})$$

Substituting equation A.8 into equation A.4 gives

$$|T(j\Omega)|^2 = \frac{1}{1 + \frac{1}{(10^{0.1\alpha_{\min}} - 1)^{-1} C_n^2\left(\frac{1}{\Omega}\right)}} = \left( 1 + \frac{1}{(10^{0.1\alpha_{\min}} - 1)^{-1} C_n^2\left(\frac{1}{\Omega}\right)} \right)^{-1}. \quad (\text{A.9})$$

From Figure 2.1, the attenuation at the passband edge frequency  $\omega_p$  (or normalized frequency  $\Omega_p$ ) is equal to  $\alpha_{\max}$ . Substituting equations A.8 and A.3 into equation A.6 with  $\Omega = \Omega_p$  yields

$$\alpha(\Omega_p) = \alpha\left(\frac{\omega_p}{\omega_s}\right) = 10 \log \left( 1 + \frac{10^{0.1\alpha_{\min}} - 1}{\cosh^2 \left[ n \cosh^{-1} \left( \frac{1}{\Omega_p} \right) \right]} \right) = \alpha_{\max}. \quad (\text{A.10})$$

Solving for the filter order  $n$  in the above equation gives

$$n = \frac{\cosh^{-1} \left[ \left( \frac{10^{0.1\alpha_{\min}} - 1}{10^{0.1\alpha_{\max}} - 1} \right)^{\frac{1}{2}} \right]}{\cosh^{-1} \left( \frac{\omega_s}{\omega_p} \right)}. \quad (\text{A.11})$$

Equation A.11 is a useful relationship for determining the required filter order  $n$  when the desired passband and stopband ranges, as well as the attenuation limits are known. It can also determine any one of the parameters knowing the other four.

To find the transfer function in the Laplace domain,  $T(s)$ , the Laplace operator  $s (=j\Omega$  for sinusoids), is substituted into equation A.4. A magnitude-squared function is the product of complex conjugate pairs. Applying this property and substituting  $\Omega = \frac{s}{j}$  in equation A.4 gives

$$|T(j\Omega)|^2 = T(s)T(-s) = \frac{KC_n^2 \left( \frac{j}{s} \right)}{1 + KC_n^2 \left( \frac{j}{s} \right)}. \quad (\text{A.12})$$

Thus, the zeros of  $T(s)$  are the roots of the equation

$$KC_n^2 \left( \frac{j}{s} \right) = 0, \quad (\text{A.13})$$

where  $K$  is defined in equation A.8, and the function  $C_n()$  is shown in equation A.3. The zeros are found as

$$z_i = \pm j \sec(\theta_i), \quad (\text{A.14})$$

where

$$\theta_i = \frac{\pi}{2n}(2i-1), \quad i = 1, 2, \dots, \frac{n}{2} \quad (\text{for even } n);$$

$$i = 1, 2, \dots, \frac{n-1}{2} \quad (\text{for odd } n). \quad (\text{A.15})$$

For stability, the poles of  $T(s)$  must lie on the left-half-side (LHS) of the complex plane. Those that lie on the right-half-side (RHS) of the complex plane correspond to an unstable system, which are not of interest and will be assigned as the poles of  $T(-s)$ . Thus, the poles of  $T(s)$  are the LHS roots of the equation

$$1 + KC_n^2 \left( \frac{j}{s} \right) = 0. \quad (\text{A.16})$$

To find these roots, the LHS roots to the equation

$$1 + KC_n^2 \left( \frac{s}{j} \right) = 0 \quad (\text{A.17})$$

are first determined. The roots to equation A.16 (the desired solutions) are the reciprocal of the roots to equation A.17. Let  $p_k$  represent a LHS complex pole pertaining to equation A.17. Separated into its real and imaginary components,  $p_k$  is given by

$$p_k = -\alpha_k + j\beta_k, \quad (\text{A.18})$$

where

$$\alpha_k = \sin(\theta_k) \sinh(a), \quad \text{and} \quad \beta_k = \cos(\theta_k) \cosh(a). \quad (\text{A.19})$$

In equation A.19,

$$a = \frac{1}{n} \sinh^{-1} \left( \frac{1}{\sqrt{K}} \right) = \frac{1}{n} \sinh^{-1} \sqrt{10^{0.1\alpha_{\min}} - 1}. \quad (\text{A.20})$$

Also,

$$\theta_k = \frac{\pi}{2n} (2k - 1), \quad k = 1, 2, 3, \dots, n. \quad (\text{A.21})$$

The final poles  $p_k^1$  (the solutions to equation A.16), are the reciprocal of  $p_k$ , given as

$$p_k^1 = \frac{1}{p_k} = \frac{1}{-\alpha_k + j\beta_k} = -\alpha_k^1 + j\beta_k^1, \quad (\text{A.22})$$

where

$$\alpha_k^1 = \frac{\alpha_k}{\alpha_k^2 + \beta_k^2} \quad \text{and} \quad \beta_k^1 = \frac{-\beta_k}{\alpha_k^2 + \beta_k^2}. \quad (\text{A.23})$$

The normalized transfer function for the Inverse-Chebyshev low-pass filter is now simply

$$T_{norm}(s) = G \frac{\prod_i [s - z_i]}{\prod_k [s - p_k^1]}, \quad (\text{A.24})$$

where  $\Pi$  represents term-by-term multiplication. The constant  $G$  represents a gain correction. For a low-pass filter,  $G \approx 1$ . Finally, the denormalized transfer function for this low-pass filter is

$$T_{den}(s) = T_{norm}(s) \Big|_{s = \frac{s}{\omega_S}}. \quad (\text{A.25})$$

## Appendix B

### *Butterworth High-Pass Response*

This derivation follows the development in Maundy (2005).

The transfer function for a low-pass Butterworth response must be found before determining the high-pass response.

The magnitude-squared function of the Butterworth response is given in its general form as

$$|T(j\omega)|^2 = \frac{1}{1 + \sum_{k=1}^n a_{2k} \omega^{2k}}, \quad (\text{B.1})$$

where  $a_{2k}$  represents a set of coefficients, and  $n$  is the order of the filter. For maximum flatness at  $\omega = 0$ , all of the coefficients for  $k \neq n$  must be zero so that the first  $(n-1)$  derivatives of equation B.1 with respect to  $\omega^2$  are zero. This leads to

$$|T(j\omega)|^2 = \frac{1}{1 + a_{2n} \omega^{2n}}. \quad (\text{B.2})$$

The loss function,  $\alpha(\omega)$ , is then

$$\alpha(\omega) = -20 \log(|T(j\omega)|) = -20 \log(1 + a_{2n} \omega^{2n})^{\frac{1}{2}} = 10 \log(1 + a_{2n} \omega^{2n}) \text{ dB}. \quad (\text{B.3})$$

The half-power angular frequency,  $\omega_o$ , is defined as the angular frequency where the loss is 3 dB. From equation B.3, this gives

$$\alpha(\omega_o) = 10 \log(1 + a_{2n} \omega_o^{2n}) = 3 \text{ dB} \approx 10 \log 2. \quad (\text{B.4})$$

Thus,

$$a_{2n} = \frac{1}{\omega_o^{2n}}. \quad (\text{B.5})$$

Substituting equation B.5 into equations B.2 and B.3 gives

$$|T(j\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_o}\right)^{2n}}, \quad (\text{B.6})$$

and

$$\alpha(\omega) = 10 \log \left( 1 + \left(\frac{\omega}{\omega_o}\right)^{2n} \right) \text{ dB}. \quad (\text{B.7})$$

From Figure 2.1, the maximum allowable attenuation in the passband,  $\alpha_{\max}$ , occurs at the passband edge angular frequency,  $\omega_p$ . Substituting this observation into equation B.7 gives

$$\alpha(\omega_p) = \alpha_{\max} = 10 \log \left( 1 + \left(\frac{\omega_p}{\omega_o}\right)^{2n} \right) \text{ dB}. \quad (\text{B.8})$$

Let a “tolerance”  $\varepsilon$  be defined as

$$\varepsilon = \sqrt{10^{0.1\alpha_{\max}} - 1}. \quad (\text{B.9})$$

Using equations B.8 and B.9 together, it can be found that

$$\omega_o = \varepsilon^{-\frac{1}{n}} \omega_p. \quad (\text{B.10})$$

Substituting equation B.10 into equation B.6 yields

$$|T(j\omega)|^2 = \frac{1}{1 + \varepsilon^2 \left(\frac{\omega}{\omega_p}\right)^{2n}}, \quad (\text{B.11})$$

and thus

$$\alpha(\omega) = 10 \log \left( 1 + \varepsilon^2 \left(\frac{\omega}{\omega_p}\right)^{2n} \right) \text{ dB}. \quad (\text{B.12})$$

Similar to the Inverse-Chebyshev case, it is useful to find a relationship between the parameters  $n$ ,  $\omega_s$ ,  $\omega_p$ ,  $\alpha_{max}$ , and  $\alpha_{min}$ . From Figure 2.1,  $\alpha_{max}$  occurs at the passband edge angular frequency,  $\omega_p$ , and  $\alpha_{min}$  occurs at the stopband edge angular frequency  $\omega_s$ . Using equation B.12 with  $\alpha(\omega_s) = \alpha_{min}$ , and solving for  $n$  with  $\varepsilon = \sqrt{10^{0.1\alpha_{max}} - 1}$  gives

$$n = \frac{\log\left(\frac{10^{0.1\alpha_{min}} - 1}{10^{0.1\alpha_{max}} - 1}\right)}{2 \log\left(\frac{\omega_s}{\omega_p}\right)}. \quad (\text{B.13})$$

To find the poles of the Butterworth function, a normalized frequency variable  $\Omega$  is first defined where

$$\Omega = \frac{\omega}{\omega_o} = \varepsilon^{\frac{1}{n}} \left( \frac{\omega}{\omega_p} \right). \quad (\text{B.14})$$

With this normalized frequency variable, define another magnitude-squared function as

$$|T(j\Omega)|^2 = \frac{1}{1 + \Omega^{2n}} = \frac{1}{|B_n(j\Omega)|^2}, \quad (\text{B.15})$$

with a corresponding loss function given by

$$\alpha(\Omega) = 10 \log(1 + \Omega^{2n}). \quad (\text{B.16})$$

In equation B.15,  $B_n(j\Omega)$  represents the Butterworth Polynomial, which is an analytic function. The Laplace parameter  $s$  can be defined as  $s=j\Omega$  for sinusoids. Making this substitution for  $|B_n(j\Omega)|$  and equating the denominators of equation B.15 while setting the result to zero gives

$$|B_n(j\Omega)|^2 = B_n(j\Omega)B_n(-j\Omega) = B_n(s)B_n(-s) = 1 + \left(\frac{s}{j}\right)^{2n} = 1 + (-s^2)^n = 0. \quad (\text{B.17})$$

Solving equation B.17 for  $s$  yields the poles of the product  $T(s)T(-s)$ . These  $s$ -poles,  $S_i$ , are

$$S_i = e^{\frac{j\pi}{2}} e^{\frac{j\pi(2i-1)}{2n}}, \quad i = 1, 2, \dots, 2n. \quad (\text{B.18})$$

Equation B.18 shows that the poles are equally spaced on a unit circle and separated by an angle of  $\frac{\pi}{n}$  radians. As with the Inverse-Chebyshev case, the poles on the left-half-side (LHS) of the complex plane pertain to  $T(s)$ . The poles on the right-half-side are not of interest, as they result in instability and pertain to  $T(-s)$ . The LHS poles,  $S_k$ , are found to be

$$\begin{aligned} S_k &= e^{j\theta_k} = -\sin \theta_k + j \cos \theta_k, \\ \theta_k &= \frac{\pi}{2n}(2k-1), \quad k = 1, 2, \dots, n. \end{aligned} \quad (\text{B.19})$$

The normalized transfer function  $T_{norm}(s)$  of a Butterworth low-pass filter is then

$$T_{norm}(s) = \frac{1}{B_n(s)} = \frac{1}{\prod_{k=1}^n (s - S_k)}. \quad (\text{B.20})$$

The denormalized transfer function for a low-pass response is

$$T(s) = T_{norm}(s) \Big|_{s=\frac{s}{\omega_o}}. \quad (\text{B.21})$$

The high-pass Butterworth response is obtained using a low-pass-prototype (LPP) algorithm. First, the half-power frequency  $\omega_o$  is found using  $\omega_o = \varepsilon^{\frac{1}{n}} \omega_p$ . Note that this is slightly different than equation B.10 which corresponds to a pure low-pass response. Then, the normalized low-pass-prototype transfer function  $T_{LPPnorm}(s)$  is found using equations

B.19 and B.20. Once  $T_{LPPnorm}(s)$  is known, the denormalized high-pass Butterworth transfer function is

$$T_{HP}(s) = T_{LPPnorm}(s) \Big|_{s=\frac{\omega_0}{s}} \cdot \quad (\text{B.22})$$

## Appendix C

### *Principal Components Analysis Mathematical Theory*

Principal components analysis (PCA) and fundamental linear algebra theory presented by Dunteman (1989), Jackson (1991), Lupton (1993), Jolliffe (2002), Smith (2002), and Shlens (2003), are discussed here in a compact and paraphrased form. This section is written for applicability to microseismic file classification discussed in section 5.4.

Assume an experiment has been performed where  $m$  variables have been measured  $n$  times. The quantity  $n$  equivalently corresponds to the number of trials in the experiment. Define an  $m \times n$  matrix  $\mathbf{A}$  containing these observations after normalizing each row by dividing row elements by the largest value (in magnitude) found in the row. Normalizing is necessary to ensure each matrix element is dimensionless, so that data can be freely projected onto principal component vectors at a later stage. Normalizing also ensures that measurements that are orders of magnitude larger than others do not drive the entire dataset when performing PCA. The matrix  $\mathbf{A}$  after data normalization can be written as

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}. \quad (\text{C.1})$$

In equation C.1,  $a_{ij}$  represents a normalized measurement of the  $i^{\text{th}}$  variable corresponding to the  $j^{\text{th}}$  trial. Thus, each row of  $\mathbf{A}$  corresponds to a single variable, and each column corresponds to a single trial.

To obtain zero-mean data, which is required to perform PCA, define a matrix  $\mathbf{B}$

where the row averages have been subtracted from  $\mathbf{A}$ . In other words,

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1n} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ b_{m1} & b_{m2} & b_{m3} & \dots & b_{mn} \end{bmatrix} = \begin{bmatrix} a_{11} - \mu_1 & a_{12} - \mu_1 & a_{13} - \mu_1 & \dots & a_{1n} - \mu_1 \\ a_{21} - \mu_2 & a_{22} - \mu_2 & a_{23} - \mu_2 & \dots & a_{2n} - \mu_2 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{m1} - \mu_m & a_{m2} - \mu_m & a_{m3} - \mu_m & \dots & a_{mn} - \mu_m \end{bmatrix}, \quad (\text{C.2})$$

where

$$\mu_i = \frac{\sum_{j=1}^n a_{ij}}{n}, \quad (i = 1, 2, \dots, m). \quad (\text{C.3})$$

The concept of covariance is required to perform PCA. Referring to equation C.2, define  $B_i$  as a single variable where the element  $b_{ij}$  represents a measurement of  $B_i$  corresponding to the  $j^{\text{th}}$  trial. Then, the covariance between two arbitrary variables  $B_x$  ( $i$  set equal to  $x$ ) and  $B_y$  ( $i$  set equal to  $y$ ) over  $n$  trials is

$$\sigma_{xy}^2 = \frac{\sum_{j=1}^n (b_{xj} - \overline{B_x})(b_{yj} - \overline{B_y})}{n-1}. \quad (\text{C.4})$$

In equation C.4,  $\sigma_{xy}^2$  is the covariance between  $B_x$  and  $B_y$ , while  $\overline{B_x}$  and  $\overline{B_y}$  are the averages of variables  $B_x$  and  $B_y$ , respectively. Referring to equation C.2, the averages of the variables have been subtracted to obtain zero-mean data for PCA, resulting in both  $\overline{B_x}$  and  $\overline{B_y}$  equaling zero. Thus, for PCA, equation C.4 reduces to

$$\sigma_{xy}^2 = \frac{\sum_{j=1}^n b_{xj} b_{yj}}{n-1}. \quad (\text{C.5})$$

In the case where  $x = y$ , equation C.5 would correspond to calculating the variance of a single variable, as opposed to calculating the covariance between two different variables.

Calculating covariance quantifies important data characteristics between two variables. A covariance of zero between two variables signifies that there is no correlation or redundancy between them. In other words, if one variable increases, it is equally likely for the other variable to either increase or decrease. The larger the calculated covariance (in magnitude) between two variables, the larger the degree of linear correlation. If a positive (negative) covariance is calculated, it is more likely that increasing one variable results in an increase (decrease) of the other, resulting in a positive (negative) linear correlation.

Figure C.1 contains sketches to give a generalized illustration of hypothetical crossplots between two arbitrary variables  $B_1$  and  $B_2$  corresponding to negligible covariance, large positive covariance, and large negative covariance.

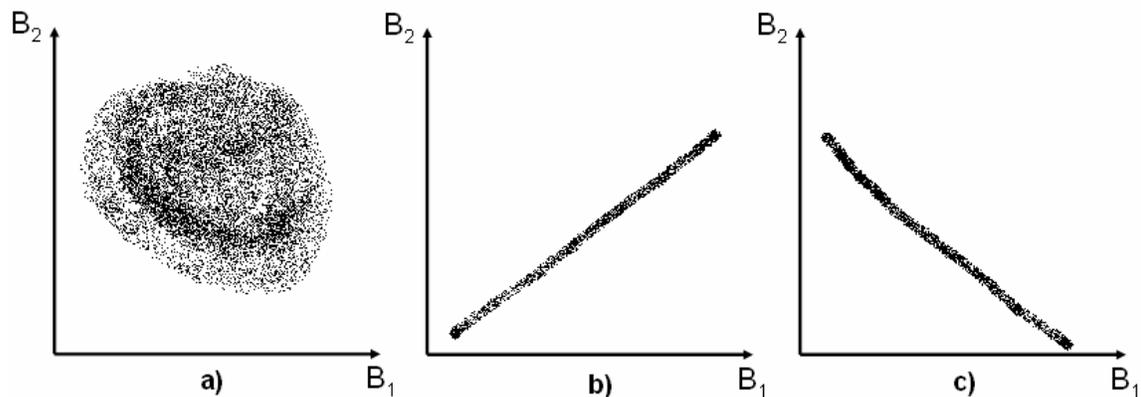


Figure C.1: Sketches of hypothetical crossplots to illustrate significance of covariance calculations between two example variables  $B_1$  and  $B_2$ . a) Negligible covariance. b) Large positive covariance. c) Large negative covariance.

Define a square “covariance matrix”  $\mathbf{C}$  that contains an exhaustive set of all possible two-variable covariance values as

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1m} \\ c_{21} & c_{22} & \cdots & c_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mm} \end{bmatrix}, \quad (\text{C.6})$$

where an arbitrary matrix element

$$c_{xy} = \sigma_{xy}^2, \quad (\text{C.7})$$

as defined in equation C.5. Thus, each off-diagonal element in  $\mathbf{C}$  represents the covariance between two variables in the dataset. Each diagonal element represents the variance of a single variable in the dataset. Referring to equations C.5, C.6, and C.7, it is clear that  $\mathbf{C}$  is symmetric ( $c_{xy} = c_{yx}$ ).

Through examining equations C.2, C.5, C.6, and C.7, the covariance matrix  $\mathbf{C}$  can be expressed through matrix multiplication as

$$\mathbf{C} = \frac{1}{n-1} \mathbf{B}\mathbf{B}^T, \quad (\text{C.8})$$

where  $\mathbf{B}^T$  is the matrix transpose of  $\mathbf{B}$ . In general, the off-diagonal elements of  $\mathbf{C}$  will not be zero, indicating a degree of correlation, and thus redundancy, between variables.

The purpose of PCA is to develop  $m$  vectors, called principal components, on which to project the data. These principal component vectors should be orthogonal to each other, resulting in no existing redundancy between data projected onto these vectors. Define an  $m \times m$  square matrix  $\mathbf{V}$  containing these developed principal component vectors as

$$\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1m} \\ v_{21} & v_{22} & \dots & v_{2m} \\ \vdots & \vdots & \dots & \vdots \\ v_{m1} & v_{m2} & v_{m3} & v_{mm} \end{bmatrix}, \quad (\text{C.9})$$

where the  $m^{\text{th}}$  row of  $\mathbf{V}$  contains the  $m^{\text{th}}$  developed principal component vector. Since the principal components are orthogonal, the vector dot product between any two principal component vectors should be zero. In other words,

$$\sum_{j=1}^m v_{rj}v_{sj} = 0, \quad (\text{C.10})$$

where  $r$  and  $s$  are integers in the range  $1 \leq r, s \leq m$  with  $r \neq s$ .

Define a matrix  $\mathbf{P}$  representing the data in  $\mathbf{B}$  projected onto the vectors in  $\mathbf{V}$ , where

$$\mathbf{P} = \mathbf{VB} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \dots & p_{1n} \\ p_{21} & p_{22} & p_{23} & \dots & p_{2n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ p_{m1} & p_{m2} & p_{m3} & \dots & p_{mn} \end{bmatrix}. \quad (\text{C.11})$$

The dimensions of  $\mathbf{P}$  are equal to the dimensions of  $\mathbf{B}$ . Similar to  $\mathbf{B}$ , the columns of  $\mathbf{P}$  correspond to trials; however, the rows of  $\mathbf{P}$  correspond to data projected onto the developed principal components. The  $i^{\text{th}}$  row of  $\mathbf{P}$  contains the resulting data projection onto the  $i^{\text{th}}$  principal component.

Let  $\mathbf{D}$  represent the covariance matrix corresponding to the projected data contained in  $\mathbf{P}$ . Similar to equation C.8,  $\mathbf{D}$  can be expressed as

$$\mathbf{D} = \frac{1}{n-1} \mathbf{PP}^T. \quad (\text{C.12})$$

The off-diagonal elements in  $\mathbf{D}$  represent covariance between data projected onto two different principal component vectors. Since the principal component vectors must be

orthogonal, this implies that all off-diagonal elements in  $\mathbf{D}$  must be zero. Thus,  $\mathbf{D}$  is a diagonal matrix. The  $i^{\text{th}}$  diagonal element in  $\mathbf{D}$  corresponds to the variance of the data after having been projected onto the  $i^{\text{th}}$  principal component vector. The principal component with the highest corresponding data variance after projection represents the direction of variation where most meaningful information can be extracted from the data.

To determine the elements of  $\mathbf{V}$ , substitute equation C.11 into equation C.12 to obtain

$$\mathbf{D} = \frac{1}{n-1} (\mathbf{V}\mathbf{B})(\mathbf{V}\mathbf{B})^{\text{T}}. \quad (\text{C.13})$$

Applying the identity  $(\mathbf{V}\mathbf{B})^{\text{T}} = \mathbf{B}^{\text{T}}\mathbf{V}^{\text{T}}$  with the substitution  $\mathbf{C} = \frac{1}{n-1} \mathbf{B}\mathbf{B}^{\text{T}}$  (equation C.8)

results in

$$\mathbf{D} = \frac{1}{n-1} \mathbf{V}\mathbf{B}\mathbf{B}^{\text{T}}\mathbf{V}^{\text{T}} = \mathbf{V}\mathbf{C}\mathbf{V}^{\text{T}}. \quad (\text{C.14})$$

As previously stated,  $\mathbf{C}$  is an  $m \times m$  symmetric and square covariance matrix of  $\mathbf{B}$ . If  $\mathbf{D}$  is to be a diagonal matrix, then the product  $\mathbf{V}\mathbf{C}\mathbf{V}^{\text{T}}$  must be a diagonal matrix. Consequently, the matrix  $\mathbf{V}$ , which contains the desired principal components, must be chosen such that  $\mathbf{V}\mathbf{C}\mathbf{V}^{\text{T}}$  is a diagonal matrix.

The principal component vectors in  $\mathbf{V}$  can be determined through examination of the symmetric and square covariance matrix  $\mathbf{C}$ . It has been shown that a symmetric matrix can be expressed through matrix multiplication of its eigenvectors with a diagonal matrix. Thus, the concept of eigenvectors and eigenvalues will first be briefly discussed.

When a matrix is multiplied by one of its eigenvectors, the result is a scaling of the eigenvector by a corresponding eigenvalue. Define  $\mathbf{E}_i$  as an  $m \times 1$  matrix representing one

of the  $m$  eigenvectors of  $\mathbf{C}$  ( $i = 1, 2, \dots, m$ ) with a corresponding eigenvalue  $\lambda_i$ . Then the relationship

$$\mathbf{C}\mathbf{E}_i = \lambda_i\mathbf{I}\mathbf{E}_i \quad (\text{C.15})$$

would be satisfied. The matrix  $\mathbf{I}$  is the identity matrix that contains values of 1 on all diagonal elements and values of 0 elsewhere. As previously stated,  $m$  corresponds to the number of measured variables in the dataset.

To obtain  $\lambda_i$  and  $\mathbf{E}_i$ , equation C.15 can be rewritten as

$$(\mathbf{C} - \lambda_i\mathbf{I})\mathbf{E}_i = 0, \quad (\text{C.16})$$

which will have non-trivial solutions only if the condition

$$|\mathbf{C} - \lambda_i\mathbf{I}| = 0 \quad (\text{C.17})$$

is satisfied. In equation C.17, the vertical bars “|” correspond to a determinant calculation. Thus, equation C.17 states that non-trivial solutions will exist only if the calculated determinant of the matrix  $(\mathbf{C} - \lambda_i\mathbf{I})$  is zero. Solving equation C.17 will yield  $m$  eigenvalues ( $\lambda_1, \lambda_2, \dots, \lambda_m$ ), from which a chosen  $\lambda_i$  can be substituted into equation C.16. Following this, a system of  $m$  equations (of which  $m-1$  of them are linearly independent) with  $m$  unknowns will be created. By setting one of the  $m$  unknowns to an arbitrary constant, the elements of  $\mathbf{E}_i$  can be determined. This procedure can be repeated  $m$  times to obtain all of the  $m$  eigenvectors of  $\mathbf{C}$  ( $\mathbf{E}_1, \mathbf{E}_2, \dots, \mathbf{E}_m$ ). Since  $\mathbf{C}$  is symmetric, these eigenvectors will be orthogonal to each other, signifying that the dot product between any two different eigenvectors will be zero. Define unit vectors  $\hat{\mathbf{E}}_i$  ( $i = 1, 2, \dots, m$ ) as

$$\hat{\mathbf{E}}_i = \frac{\mathbf{E}_i}{\|\mathbf{E}_i\|}, \quad (\text{C.18})$$

where  $\|\mathbf{E}_i\|$  represents the length of vector  $\mathbf{E}_i$ .

Define a matrix  $\mathbf{F}$  that contains all of the  $m$  unit-length eigenvectors of  $\mathbf{C}$  ( $\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2 \dots \hat{\mathbf{E}}_m$ ) with each unit-length eigenvector placed in a single column. Thus,  $\mathbf{F}$  will be a square  $m \times m$  matrix whose columns are orthogonal to each other. To obtain  $\mathbf{V}$ , which contains the desired principal component vectors, further examination of the symmetric matrix  $\mathbf{C}$  is required. As previously stated, it has been shown that a symmetric and square matrix can be expressed through matrix multiplication of its orthogonal eigenvectors with a diagonal matrix. Specifically, it has been shown that a symmetric matrix, such as  $\mathbf{C}$ , can be expressed as

$$\mathbf{C} = \mathbf{F}\mathbf{H}\mathbf{F}^T, \quad (\text{C.19})$$

where  $\mathbf{H}$  is a diagonal matrix. Substituting equation C.19 into equation C.14 gives

$$\mathbf{D} = \mathbf{V}\mathbf{F}\mathbf{H}\mathbf{F}^T\mathbf{V}^T. \quad (\text{C.20})$$

It is desired to choose  $\mathbf{V}$  such that only the diagonal matrix  $\mathbf{H}$  remains on the right-hand side of equation C.20, resulting in  $\mathbf{D}$  being a diagonal matrix and thus implying that the projected data contained in  $\mathbf{P}$  has zero covariance. This would further imply that redundancy is eliminated in  $\mathbf{P}$ . It can be easily demonstrated that, since  $\mathbf{F}$  is a square matrix whose columns have unit length and are orthogonal to each other (the dot product between any two different columns will be zero), the relationship

$$\mathbf{F}^T\mathbf{F} = \mathbf{I} \quad (\text{C.21})$$

will be satisfied. In equation C.21, the element in the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of  $\mathbf{F}^T\mathbf{F}$ , where  $i, j = 1, 2 \dots m$  simply corresponds to calculating the dot product of columns  $i$  and  $j$  in the matrix  $\mathbf{F}$ . Thus, the diagonal elements of  $\mathbf{F}^T\mathbf{F}$ , which are dot products of single columns

with themselves, represent the column lengths, which are unity. The off-diagonal elements of  $\mathbf{F}^T \mathbf{F}$  will be zero, since columns of  $\mathbf{F}$  are orthogonal with respect to each other. This confirms the relationship in equation C.21.

Comparing equations C.20 and C.21, setting

$$\mathbf{V} = \mathbf{F}^T \quad (\text{C.22})$$

would be appropriate. This results in

$$\mathbf{D} = \mathbf{V} \mathbf{F} \mathbf{H} \mathbf{F}^T \mathbf{V}^T = (\mathbf{F}^T \mathbf{F}) \mathbf{H} (\mathbf{F}^T \mathbf{F}) = \mathbf{I} \mathbf{H} \mathbf{I} = \mathbf{H}, \quad (\text{C.23})$$

which is desired, as this results in  $\mathbf{D}$  becoming a diagonal matrix implying that the projected data contained in  $\mathbf{P}$  has zero covariance. This further implies that redundancy is eliminated in  $\mathbf{P}$ .

The results obtained suggest that the principal components of the dataset contained in the rows of  $\mathbf{V}$  are equal to the unit-length eigenvectors of  $\mathbf{C}$  contained in the columns of  $\mathbf{F}$ . Thus, the principal components of the dataset are found by determining the unit-length eigenvectors ( $\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2, \dots, \hat{\mathbf{E}}_m$ ) of the covariance matrix  $\mathbf{C}$ .

In summary,  $\mathbf{B}$  contains mean-corrected measurements of several variables that could be correlated and thus redundant to a degree. This redundancy can be seen through non-zero off-diagonal elements in  $\mathbf{C}$ , where  $\mathbf{C}$  is a covariance matrix corresponding to the measurements in  $\mathbf{B}$ . It is desired to project the data in  $\mathbf{B}$  onto orthogonal principal component vectors contained in the rows of  $\mathbf{V}$ , eliminating redundancy. This projected data is stored in the matrix  $\mathbf{P}$ . Each principal component present in each row of  $\mathbf{V}$  is a single unit-length eigenvector of the covariance matrix  $\mathbf{C}$ .

It has been shown in the cited literature that data variance after projection onto a principal component, where each principal component is a single eigenvector, is equal to

the eigenvalue of the corresponding eigenvector. The eigenvector whose eigenvalue is largest corresponds to the direction of maximum variance in the dataset. Data projected onto this eigenvector will yield the most useful information about the dataset. The eigenvector whose eigenvalue is second largest is oriented in the direction of maximum possible data variance given the restriction that it must be orthogonal to the first eigenvector. A similar relationship would hold for the third “most important” eigenvector relative to the second, and so on. Eigenvectors whose eigenvalues are relatively small likely correspond to noise components, where little meaningful information can be extracted.

For discussions pertaining to PCA application that follow,  $\hat{\mathbf{E}}_1$  will represent the principal component whose eigenvalue is largest (the first principal component). Thus, data projected onto  $\hat{\mathbf{E}}_1$  will contain maximum variance and thus yield the most meaningful information. Expressed mathematically,

$$\lambda_1 > \lambda_2 > \dots > \lambda_m. \quad (\text{C.24})$$

Figures C.2 and C.3 are hypothetical illustrations to demonstrate PCA applied to a simple two dimensional example. Assume that measurements have been performed corresponding to two arbitrary variables with zero-mean correction applied. As per convention, the first variable is plotted with respect to the  $x$  axis, while the second is plotted with respect to the  $y$  axis. A crossplot of these hypothetical measurements is shown in Figure C.2, where each circle represents a single trial. The  $x$  and  $y$  axes are oriented horizontally and vertically, respectively. Clearly, there is a strong positive linear correlation between the two variables. PCA would project the data onto two new axes corresponding to the calculated principal component vectors, as shown in Figure C.3. Only the  $\hat{\mathbf{E}}_1$  axis,

however, contains significant data variance. The other,  $\hat{\mathbf{E}}_2$ , is mainly a noise component. Thus, PCA has reduced the effective dimensionality of the data from two to one.

When analyzing data in two dimensions, it is relatively easy to determine, even visually, the direction where significant data variance exists. As data dimensionality increases, however, it becomes significantly more difficult to visualize the dataset and determine directions of maximum variance. Thus, PCA becomes an important tool to extract key information from datasets with many measured variables.

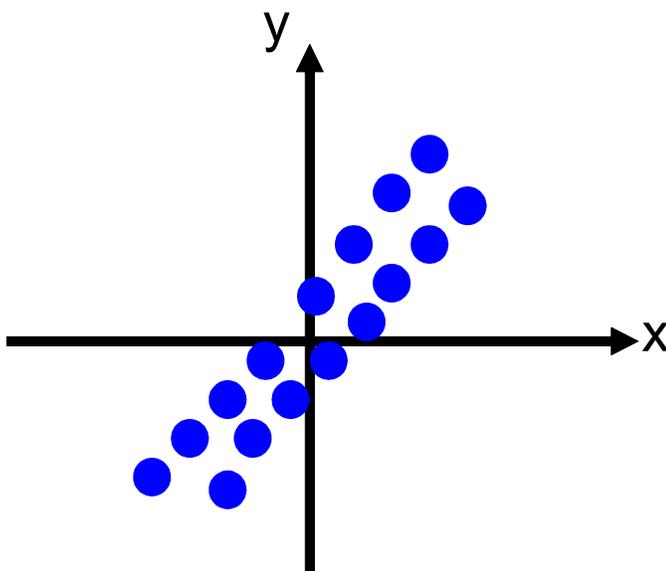


Figure C.2: Illustrative 2-D crossplot example of measurements taken for two variables with zero-mean correction applied. The first variable is plotted with respect to the  $x$  axis, and the second is plotted with respect to the  $y$  axis. Strong positive correlation is seen between the two variables.

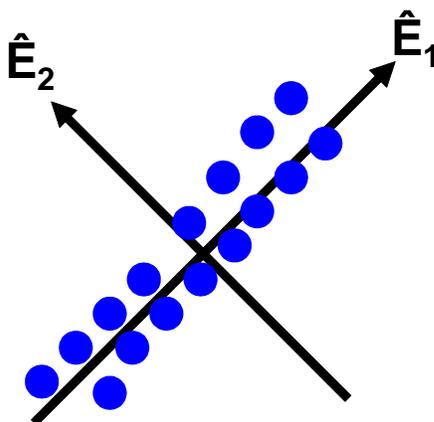


Figure C.3: Illustrative 2-D crossplot example from Figure C.2 with principal component vectors  $\hat{E}_1$  and  $\hat{E}_2$  shown. Maximum data variance is seen in the direction of  $\hat{E}_1$ , while mainly noise is seen in the direction of  $\hat{E}_2$ . Thus, data projected onto  $\hat{E}_1$  would yield virtually all important information about the dataset, resulting in the effective dimensionality of the data reduced from two to one.

**REFERENCES**

- Aki, K., and Richards, P.G., 2002, Quantitative Seismology, 2<sup>nd</sup> edition: University Science Books.
- Allen, R.V., 1997, Automatic Earthquake Recognition And Timing From Single Traces: Bulletin of the Seismological Society of America, **68**, 1521-1532.
- Ambuter, B.P., and Solomon, S.C., 1974, An event-recording system for monitoring small earthquakes: Bulletin of the Seismological Society of America, **64**, 1181-1188.
- Campbell, G., 2005, Velocity Model Improvements for use in Imperial Oil's Microseismic Analysis Program: Imperial Oil Resources & University of Alberta.
- Chael, E.P., 1997, An Automated Rayleigh-Wave Detection Algorithm: Bulletin of the Seismological Society of America, **87**, 157-163.
- Correig, A.M., and Urquizú, M., 2002, Some dynamical characteristics of microseism time-series: Geophysical Journal International, **149**, 589-598.
- DeNoyer, J.M., 1966, S waves generated by small seismic sources: Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, **290**, 448-460.
- Duncan, P.M., 2005, Is there a future for passive seismic?: First Break, **23**, 111-115.
- Dunteman, G.H., 1989, Principal Components Analysis: Sage Publications.
- Eastwood, J., 1993, Temperature-dependent propagation of P- and S-waves in Cold Lake oil sands: Comparison of theory and experiment: Geophysics, **58**, 863-872.
- Engdahl, E.R., Van der Hilst, R., and Buland, R., 1998, Global Teleseismic Earthquake Relocation with Improved Travel Times and Procedures for Depth Determination: Bulletin of the Seismological Society of America, **88**, 722-743.
- Feichtinger, H.G., and Strohmer, T., 1998, Gabor Analysis and Algorithms: theory and applications: Birkhäuser.
- Freedman, D., and Diaconis, P., 1981, On the histogram as a density estimator:  $L_2$  theory: Probability Theory and Related Fields, **57**, 453-476.
- Haykin, S., 2001, Communication Systems, 4<sup>th</sup> edition: John Wiley & Sons.
- Haykin, S., and Van Veen, B., 2003, Signals and Systems, 2<sup>nd</sup> edition: John Wiley & Sons.

- He, K., and Meeden, G., 1997, Selecting the Number of Bins in a Histogram: A Decision Theoretic Approach: *Journal of Statistical Planning and Inference*, **61**, 49-59.
- Heisenberg, W., 1927, Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik: *Zeitschrift für Physik*, **43**, 172-198.
- Horiuchi, S., Negishi, H., Abe, K., Kamimura, A., and Fujinawa, Y., 2005, An Automatic Processing System for Broadcasting Earthquake Alarms: *Bulletin of the Seismological Society of America*, **95**, 708-718.
- Imperial Oil Ltd., 2006a, Alberta's Oil Sands: [http://www.limperiale.ca/Canada-English/Investors/Operating/Natural\\_Resources/I\\_O\\_NaturalResourcesFig1.asp](http://www.limperiale.ca/Canada-English/Investors/Operating/Natural_Resources/I_O_NaturalResourcesFig1.asp), internet web page. Accessed November 3, 2006.
- Imperial Oil Ltd., 2006b, Proposed Development Areas – Mahihkan North and Nabiye: [http://www.limperiale.ca/Canada-English/Investors/Operating/Natural\\_Resources/I\\_O\\_NaturalResourcesFig4.asp](http://www.limperiale.ca/Canada-English/Investors/Operating/Natural_Resources/I_O_NaturalResourcesFig4.asp), internet web page. Accessed November 5, 2006.
- Imperial Oil Ltd, 2006c, Cyclic Steam Stimulation - not to scale: [http://www.limperiale.ca/CanadaEnglish/Investors/Operating/Natural\\_Resources/I\\_O\\_NaturalResourcesFig5.asp](http://www.limperiale.ca/CanadaEnglish/Investors/Operating/Natural_Resources/I_O_NaturalResourcesFig5.asp), internet web page. Accessed November 5, 2006.
- Isaac, H.J., 1996, Seismic Methods for Heavy Oil Reservoir Monitoring: PhD. thesis, University of Calgary.
- Jackson, J.E., 1991, *A User's Guide to Principal Components*: John Wiley & Sons.
- Jolliffe, I.T., 2002, *Principal Component Analysis*, 2<sup>nd</sup> edition: Springer.
- Joswig, M., 1990, Pattern Recognition For Earthquake Detection: *Bulletin of the Seismological Society of America*, **80**, 170-186.
- Lange, K., 2003, *Applied Probability*: Springer.
- Lay, T., and Wallace, T.C., 1995, *Modern Global Seismology*: Academic Press.
- Lee, W.H.K., and Stewart, S.W., 1981, *Principles and Applications of Microearthquake Networks*: Academic Press.
- Lupton, R., 1993, *Statistics In Theory and Practice*: Princeton University Press.
- Margrave, G.F., 2007, *Methods of Seismic Data Processing*. Geophysics 557 Course Lecture Notes, Winter 2007: The CREWES Project, University of Calgary.

- Maundy, B., 2005, ENEL 559 Course Notes, University of Calgary.
- Maxwell, S.C., and Urbancic, T.I., 2001, The role of passive microseismic monitoring in the instrumented oil field: *The Leading Edge*, **20**, 636-639.
- Miller, S.L., and Childers, D.G., 2004, *Probability and Random Processes With Applications to Signal Processing and Communications*: Elsevier.
- Mitzenmacher, M., and Upfal, E., 2005, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*: Cambridge University Press.
- Munro, K. A., 2005, Analysis of microseismic event picking with applications to landslide and oil-field monitoring settings: MSc. thesis, University of Calgary.
- Scholz, C.H., 1990, *The Mechanics of Earthquakes and Faulting*: Cambridge University Press.
- Scott, D.W., 1979, On optimal and data-based histograms: *Biometrika*, **66**, 605-610.
- Scott, D.W., 1992, *Multivariate Density Estimation: Theory Practice, and Visualization*: John Wiley & Sons.
- Sheriff, R.E., and Geldart, L.P., 1995, *Exploration Seismology*, 2nd edition: Cambridge University Press.
- Shlens, J., 2003, *A Tutorial On Principal Component Analysis: Derivation, Discussion, and Singular Value Decomposition*, University of California, San Diego.
- Smith, L.I., 2002, *A tutorial on Principal Components Analysis*, University of Otago, New Zealand.
- Stockwell, R.G., Mansinha, L., and Lowe, R.P., 1996, Localization of the Complex Spectrum: The *S* Transform: *IEEE Transactions On Signal Processing*, **44**, 998-1001.
- Sturges, H., 1926, The choice of a class-interval: *Journal of The American Statistical Association*, **21**, 65-66.
- Suhov, Y., and Kelbert, M., 2005, *Probability and Statistics by Example*: Cambridge University Press.
- Talebi, S., Nechtschein, S., and Boone, T.J., 1998, Seismicity and Casing Failures Due to Steam Stimulation in Oil Sands: *Pure and Applied Geophysics*, **153**, 219-233.

- Tan, J.F., Bland, H.C., and Stewart, R.R., 2006, Passive seismic reservoir monitoring techniques applied to heavy oil production: CREWES Research Report, **18**, University of Calgary.
- Therrien, C.W., and Tummala, M., 2004, Probability for Electrical and Computer Engineers: CRC Press.
- Walpole, R.E., Myers, R.H., Myers, S.L., and Ye, K., 2002, Probability & Statistics For Engineers & Scientists, 7<sup>th</sup> edition: Prentice Hall.
- Warren, L.M., and Shearer, P.M., 2006, Systematic determination of earthquake rupture directivity and fault planes from analysis of long-period *P*-wave spectra: Geophysical. Journal International, **164**, 46–62.
- Wheeler, J.A., and Zurek, W.H., 1983, Quantum Theory and Measurement: Princeton University Press.
- Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S., and Trujillo, J., 1998, A Comparison of Select Trigger Algorithms for Automated Global Seismic Phase and Event Detection: Bulletin of the Seismological Society of America, **88**, 95-106.
- Zhou, H., and McMechan, G.A., 1999, Parallel Butterworth and Chebyshev dip filters with applications to 3-D seismic migration: Geophysics, **64**, 1573-1578.